Guided Learning Center

IBM System/38 Implementation Topics Student Materials Book

ZR30-1020-3

IBM System/38 Implementation Topics Student Materials Book

Preface

System/38 Implementation Topics is a multivolume set of books designed for students who will implement an IBM System/38. This Student Materials Book provides the student with a course overview, an introduction and summary for each module, module exercises, and sections for notes about each part of the course.

This entire set of books is designed to be used in a Guided Learning Center environment supplemented by audiovisual material and reference manuals. Exercises are included in most modules to be executed on the Guided Learning Center's System/38.

Module 1 System Installation
Module 2 Security
Module 3 Work Management
Module 4 Control Language Programming
Module 5 Source Entry Utility Additional Topics
Module 6 Message Handling
Module 7 Data Base Topics
Module 8 Save/Restore

Fourth Edition (November 1986)

This is a major revision of, and obsoletes, ZR30-1020-2. Changes are periodically made to the information herein; any such changes will be reported in subsequent editions.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

You may communicate your comments about this publication, its organization or subject matter with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you. Address your comments to IBM Corporation, Department 7TF, P.O. Box 2150, Atlanta, Georgia 30055.

Contents

Section I

Overview of Implementation Topics

In the installation of your System/38, you need to integrate many different tasks to produce a smooth running system. You need to be able to:

- Install the System/38 to meet your requirements
- Control the operation of the System/38
- Execute all your different jobs in a controlled, organized manner
- Utilize System/38 functions to support your applications
- Prepare for and recover from loss of information

And, you must address each of these needs in a planned fashion to make the best use of your resources. This is called implementation. Implementation is the blending of all your previous design, application development, and coding activities into an efficient system to meet your data processing needs.

Each of the modules in this course covers different aspects of **system implementation.** Taken together, they form an installation plan for your applications on the System/38. The modules describe the tasks you must address as an implementor and the System/38 tools available to assist you. The modules that make up this course are:

- Module 1 . . . System Installation
- Module 2 . . . Security
- Module 3 . . . Work Management
- Module 4 . . . Control Language Programming
- Module 5 . . . Source Entry Utility Additional Topics
- Module 6 . . . Message Handling
- Module 7 . . . Data Base Topics
- Module 8 . . . Save/Restore

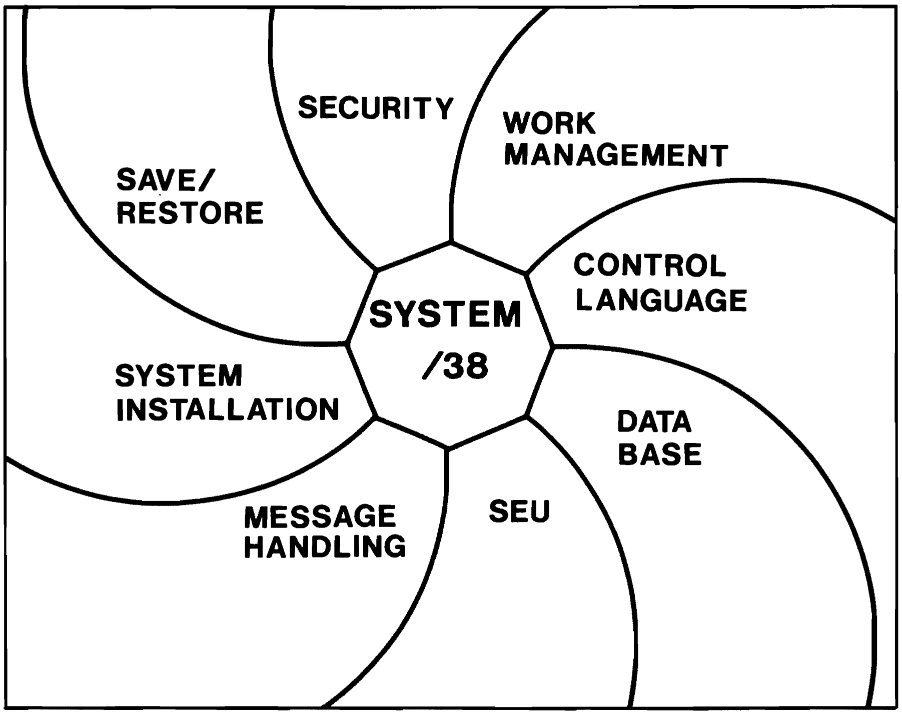
The topics are arranged in the order that you, the implementor, might address them.

Implementation Considerations

The primary objective of your System/38 education to date has been to prepare you to code, test, and execute application programs. However, you need more than just programs to have a complete, usable application. In addition to the application programs, you need system functions and operating procedures.

The Implementation Topics course defines these additional pieces and ties them together. It is planning how all these pieces will operate on a daily basis. Implementation is the integration of your application programs and the functions of the System/38 in a planned fashion for efficient daily operation.

However, just what needs to be planned? What system functions and operating capabilities are available and how are they tied together with your application programs? In other words, what needs to be implemented? Your system implementation considerations are the major topics of this course.



- Installation . . . Before any application can run, your System/38 must be installed. You must tailor the System/38 to meet your specific needs. Since you must install your devices and program products before you can use your System/38, system installation is your first implementation consideration.

- Security . . . Once your System/38 is installed, you need to be able to control who has access to what and perhaps limit the functions a user can perform. The security module introduces you to the aids provided by the System/38 to help you prevent unauthorized access to your system and objects.

- Work Management . . . How does a job flow within the system from initiation to completion? How can you control that flow? The work management function of CPF controls how jobs are handled within the system. The work management module identifies the elements of work management and how you can implement them to meet your needs.

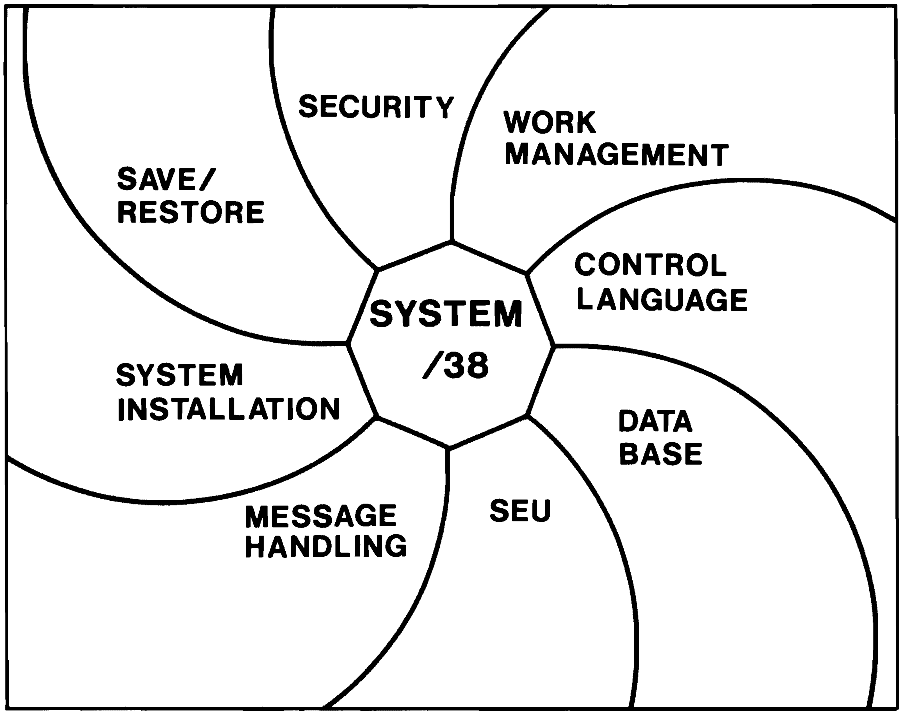
- Control Language Programming . . . To facilitate the execution of jobs, you can build a menu of desired items from which jobs are selected by number. Also, you need to build in techniques to check for and perhaps change system conditions as a job is executing. Control language and control language programs give you these capabilities, and more. Module 4 shows you how to implement these functions.

- SEU Additional Topics . . . As you code high level language programs and control language programs, you need to be able to enter them into your System/38 quickly and easily. You used SEU in your earlier System/38 education. This module adds to your knowledge of SEU, making it an even more useful implementation tool.

- Message Handling . . . You may have to communicate with another user. During the execution of different jobs, the system may have a message for you, or a program may send or receive a message. The message handling module covers communication between you and the System/38 and how to implement different types of message processing.

- Data Base Topics . . . You were introduced to data base topics in your earlier System/38 education and are aware of various characteristics of data base files. This module expands on these topics by addressing implementation considerations for file creation and maintenance. It also introduces some new System/38 functions you can implement to control your data base files.

- Save/Restore . . . Finally, after your system is installed, your application and control language programs are ready, and your data base files are in place, you need a way of protecting them from loss or damage. The last module shows you the System/38 functions you can implement to prepare for and recover from loss or damage to your system and its information.



Implementation Reference Manuals

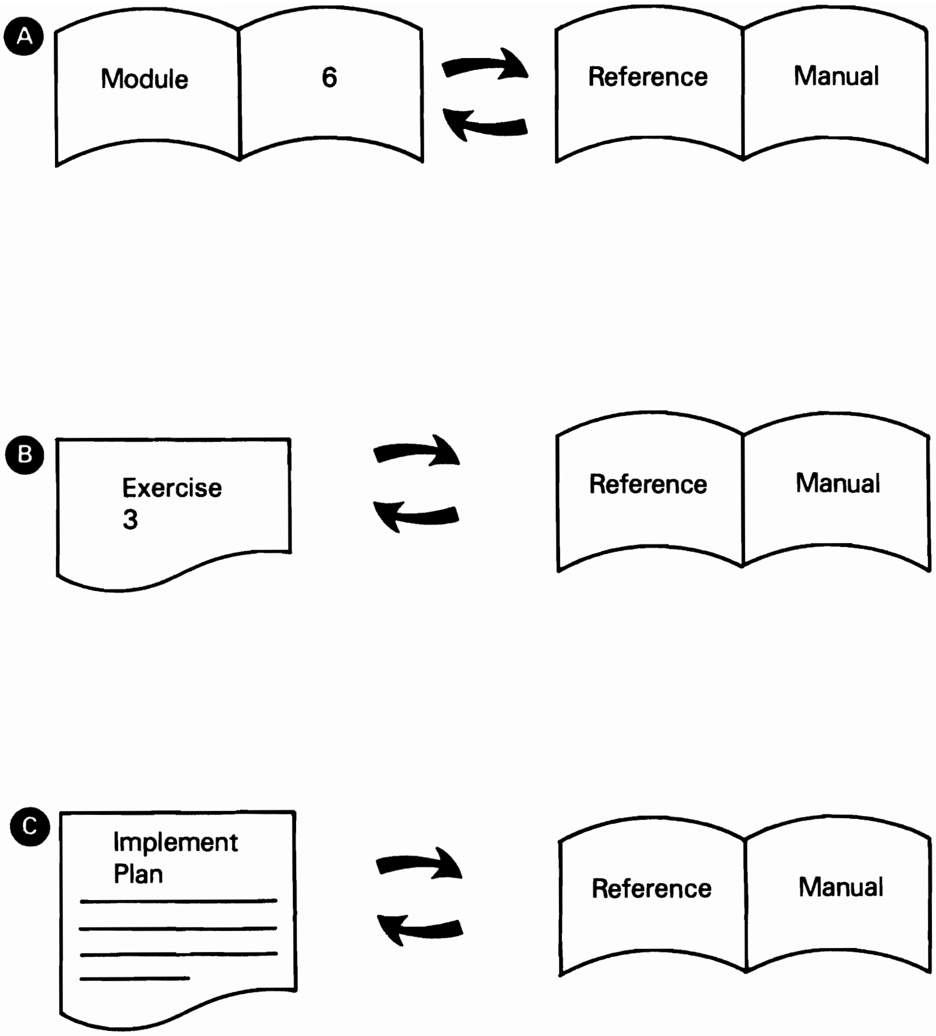
As you can imagine, all the implementation topics are explained in various System/38 reference manuals. However, where do you begin to learn all this material? The course modules are your starting point. But, rather than duplicating the reference manuals in the course modules, the module texts guide you through the material and direct you to appropriate sections of the reference manuals. The modules introduce you to the topics and give you a basic understanding of them. The reference manuals provide the detail.

You will use several reference manuals during this course and later on in your implementation efforts. They are:

- Source Entry Utility Reference Manual SC21-7722
- Control Program Facility Programmer's Guide SC21-7730
- Control Language Reference Manual SC21-7731
- Programming Reference Summary SC21-7734
- Operator's Guide SC21-7735
- Guide to Program Product Installation and Device Configuration GC21-7775
- Data Description Specifications manual SC21-7806
- Problem Determination Guide SC21-7876
- 5250 Information Display System Planning and Site Preparation Guide GA21-9337
- IBM 5292 Color Display Station Setup Procedure (GA21-9415)

Each of these manuals is located in the Guided Learning Center Resource Library. Instructions in the module texts direct you to the proper reference manual for the topic you are studying.

You may wish to do extra reading on a topic or another related topic at any time. To help you find the many references to each System/38 subject, use the Guide to Publication (Master Index section), GC21-7726.



Reference Manual Usage

Reference Manual Usage

As you have just read, you will use a number of specific manuals throughout this course and, more importantly, when you actually begin your implementation efforts. Each manual has its own Table of Contents and an index to help you locate specific information. When the module text directs you to locate a specific topic, you should use the Table of Contents or index to do so. The three ways you should use the reference manuals are diagrammed on the facing page.

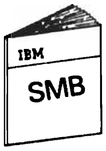
Ⓐ While you are studying a module, a topic is introduced and related to the overall implementation effort. You may then be directed to a section of a reference manual for coding rules or specific details about that topic. You should consider making a note in your Student Materials Book indicating the reference manual location of the information.

Ⓑ When you have an exercise to code and want to verify your solution, you can check the rules and examples in the appropriate reference manual. For example, you may quickly check Control Language commands in the Programming Reference Summary or research them in detail in the Control Language Reference Manual.

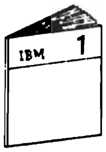
Ⓒ You should not expect to remember everything you study in this course or every coding option for every command. After this course is completed and you start to implement your own applications on your System/38, these same reference manuals guide you in completing the necessary coding or implementation activity.

You should practice using these different reference manuals now as you study each module in this course. Become familiar with them now and they will help you in the future.

Course Structure

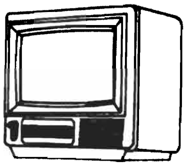


This Student Materials Book is yours to keep when you complete the course. It contains a course overview section – the section you are reading now – a Prerequisite Review section, and an Implementation Notes section. The Implementation Notes section contains the Introduction, Progress Checks, Machine Exercise instructions, Personal Notes space, and Summary for each module in the course. Since you are asked not to write in the modules texts and they are collected by the Administrator, you should record any notes, comments, thoughts, or exercise coding here, in your Student Materials Book. You should begin and end your study of each module in your Student Materials Book.



As you already know, this course is divided into modules that represent your major implementation topics. To further assist you in studying the material, each module is divided into sections of study called units. Each unit discusses a piece of the major topic. Several modules have video presentations and machine exercises and most units end with a progress check.

Most modules have a video presentation about the implementation topic. The purpose of a video is to introduce the entire topic or focus your study on one portion of a topic. For example, the system installation video shows you how to connect work stations to your system unit using planning sheets you study in the module text.

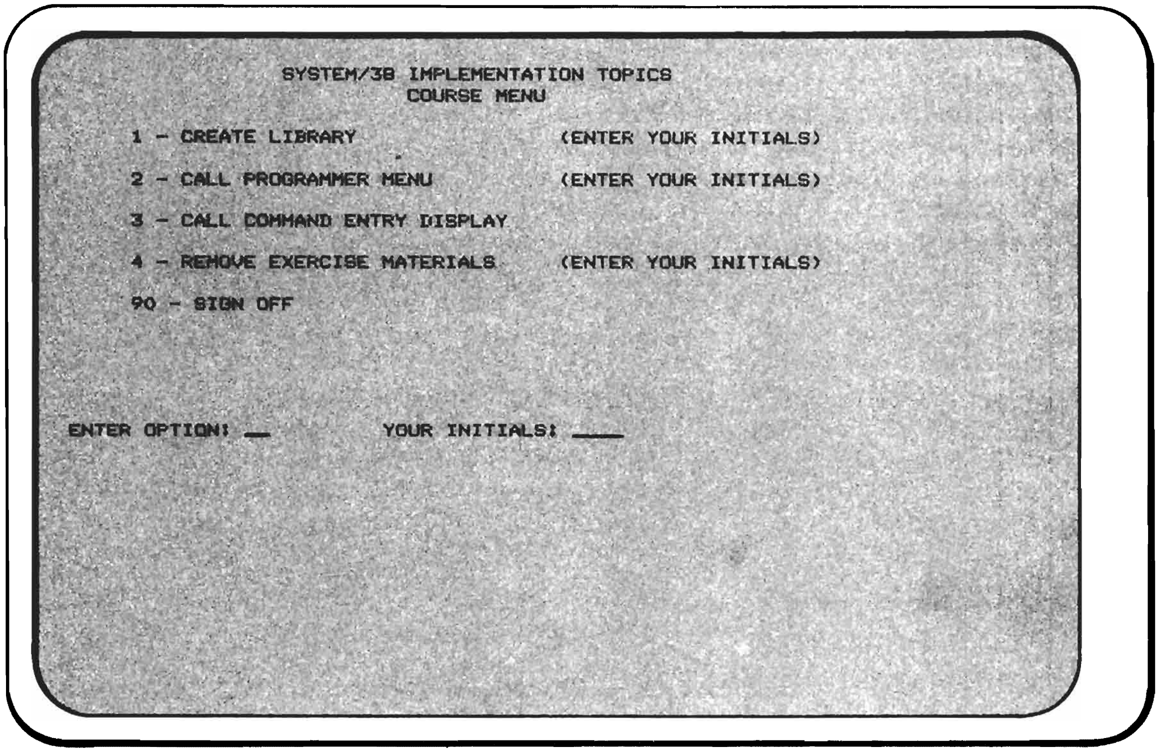


Whenever a module has a video presentation, the symbol on the left is displayed and the text directs you to the proper videodisc. All Implementation Topics presentations are stored in the Guided Learning Center library. You can replay any video segment whenever you wish. For example, you may replay a video at the end of a module as a review of the topic.

Progress Checks at the end of most units are self-evaluated questions to let you know how you are doing. They are identified with the symbol shown on the left. The questions are located in the Implementation Notes section of your Student Materials Book. You should answer these questions based on what you have studied in the current unit. Answers to the questions are located in the module. You may, if you wish, use any of the reference manuals and review any module text material during the progress checks.

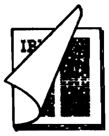
Most of the modules have one or more machine exercises to give you practice using some of the implementation tools you study. Directions in the module text tell you when to perform a machine exercise. The actual instructions for each exercise are here, in your Student Materials Book. The symbol to the left identifies a machine exercise. These exercises are run on a display station in the Guided Learning Center. Be sure to read all the instructions carefully **before** attempting to do the exercise at a display station. Do this so you know what is expected of you before you begin.

The first step of each exercise instructs you to sign on to a work station using the password **S38IMPL.** It is an abbreviation for **S**ystem/**38** **IMPL**ementation Topics, the title of this course. This password accesses a user profile established especially for Implementation Topics. When you successfully sign on, the System/38 Implementation Topics Course Menu, shown below, appears on the screen.



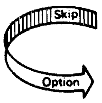
```
                SYSTEM/38 IMPLEMENTATION TOPICS
                          COURSE MENU

     1 - CREATE LIBRARY                (ENTER YOUR INITIALS)

     2 - CALL PROGRAMMER MENU          (ENTER YOUR INITIALS)

     3 - CALL COMMAND ENTRY DISPLAY

     4 - REMOVE EXERCISE MATERIALS     (ENTER YOUR INITIALS)

    90 - SIGN OFF




  ENTER OPTION: __          YOUR INITIALS: ___
```

Further instructions in each module exercise tell you what to do next. Be sure to enter your initials for the options that ask for them. (Use the initials for your first, middle, and last name. If you do not have a middle name, just use your first and last name initials.) Your initials are used to identify your individual library.



As you have read, you have reference manual reading assignments in several of the modules. These assignments are identified with the symbol to the left. Again, consider making a note of the reference manual topic, chapter and page number in your Student Materials Book.

The Programming Reference Summary is a very useful quick reference when you are coding Control Language commands. Use it as a reminder of the parameters needed for the various commands. You should have a copy of it to use while taking the course.



If you are already familiar with a certain function or are not implementing a particular feature on your system, you may skip the part of the text that discusses it. For example, if you do not have magnetic tape as part of your system, you can skip the discussion of magnetic tape in the Save/Restore module. Skipping any part of a module is entirely up to you. Skip options in a module are identified by the symbol on the left.

Section II

Prerequisite Review

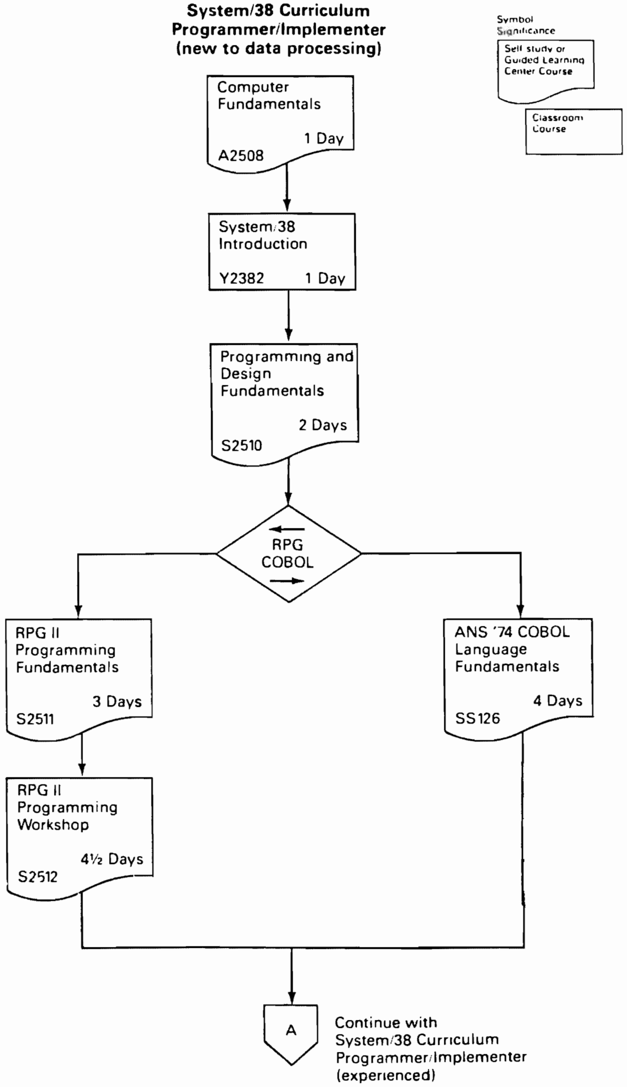
This course is for the person who will implement applications on an IBM System/38. As indicated in the flowchart on the following pages, you should have successfully completed the System/38 Application Programming course (S2585) before taking this course. Optionally, you may have completed either or both the RPG III and Structured Programming (S2584) and/or the System/38 COBOL Workshop.

This course assumes you have completed the prerequisite education. Topics addressed in prior education are referenced and, in many cases, expanded in this System/38 Implementation Topics course. In Appendix A in the back of this Student Materials Book is a brief overview of the System/38 curriculum in the Guided Learning Center. If you wish, you can refer to Appendix A to locate and study specific System/38 topics from any of the courses listed.

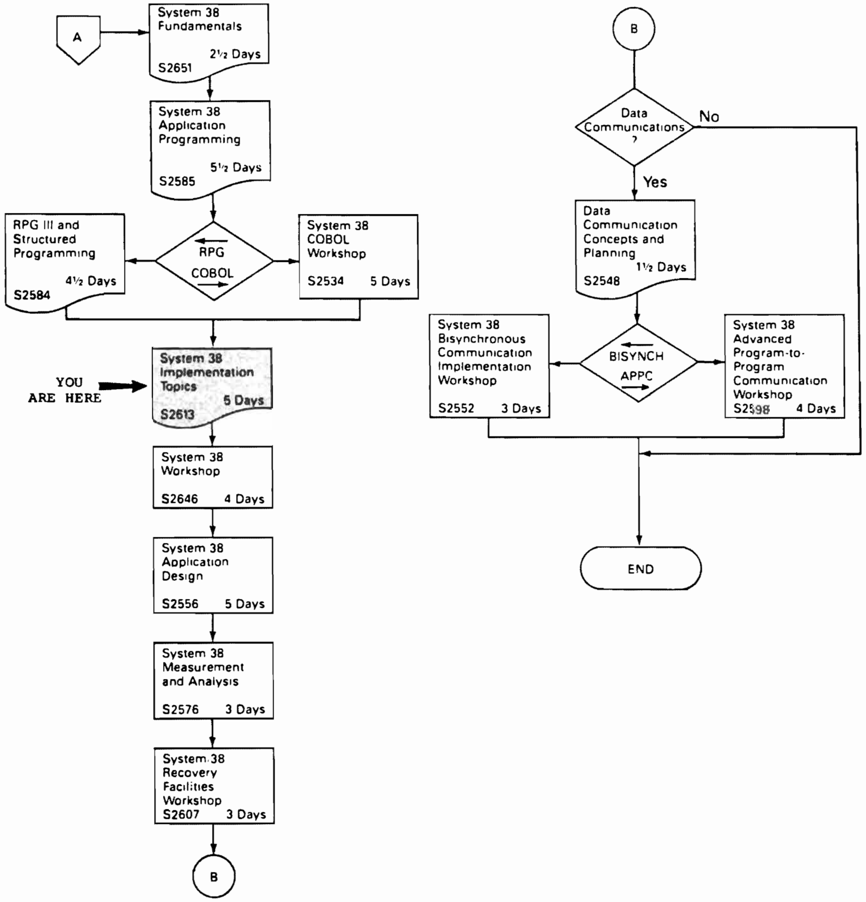
The rest of this section is a review of things you learned in earlier courses. It covers items you should already be aware of and be prepared to use as you progress through Implementation Topics. As appropriate, a reference source is provided if you need to study a topic again before you begin Implementation Topics. The review questions are divided into four parts. The answers for each part begin on the page following the questions. Read each question carefully and record your answer in the space provided.

This flowchart is provided as a guide to the educational offerings for the System/38.

System/38 Curriculum Programmer/Implementer (new to data processing)


Symbol Significance
Self study or Guided Learning Center Course
Classroom Course
Computer Fundamentals
1 Day
A2508
System/38 Introduction
Y2382
1 Day
Programming and Design Fundamentals
2 Days
S2510
RPG
COBOL
RPG II Programming Fundamentals
3 Days
S2511
RPG II Programming Workshop
4½ Days
S2512
ANS '74 COBOL Language Fundamentals
4 Days
SS126
A
Continue with System/38 Curriculum Programmer/Implementer (experienced)

System/38 Curriculum Programmer/Implementer (experienced)


A
System 38 Fundamentals
2½ Days
S2651
System 38 Application Programming
5½ Days
S2585
RPG
COBOL
RPG III and Structured Programming
4½ Days
S2584
System 38 COBOL Workshop
S2534
5 Days
YOU ARE HERE
System 38 Implementation Topics
5 Days
S2613
System 38 Workshop
S2646
4 Days
System 38 Application Design
S2556
5 Days
System 38 Measurement and Analysis
S2576
3 Days
System 38 Recovery Facilities Workshop
S2607
3 Days
B
B
Data Communications ?
No
Yes
Data Communication Concepts and Planning
1½ Days
S2548
BISYNCH
APPC
System 38 Bisynchronous Communication Implementation Workshop
S2552
3 Days
System 38 Advanced Program-to-Program Communication Workshop
S2598
4 Days
END

Part I. General Topics

1. List at least four types of objects.

___

___

___

___

2. What is a library?

___

___

3. What is a control language program?

___

___

4. Briefly describe interactive processing and batch processing.

___

___

___

___

___

___

5. What is a qualified object name?

___

___

6. How do you identify a CL variable name in a CL command?

___

___

7. What is a library list? How do you change a library list?

___

___

___

___

Review your answers with those on the following pages.

Part I. General Topics — Answers and Discussion

1. Your list should contain any four from the list below.

   - Programs
   - Job queues
   - Output queues
   - Libraries
   - Commands
   - Job descriptions
   - Files
   - User profiles
   - Data areas

   An object is anything that exists in and occupies space in storage and on which operations can be performed.

   (The Control Language Reference Manual chapter titled "Summary of CPF Functions and Object Types" contains a list of System/38 object types.)

2. A library is an object that serves as a directory to other objects.

3. A control language, or CL, program is an executable object that is created from source consisting entirely of control language commands.

   A CL program executes more quickly than uncompiled commands because it does not need to be translated into machine instructions each time it is used. One common use of CL programs is to support menus for your applications.

   (See System/38 Application Programming, Module 15, Unit 2.)

4. Interactive processing involves interaction between the user and the system. A System/38 interactive job is everything you do from the time you sign on at a work station until you sign off.

   Batch processing involves using one or more programs to process a group of data records. Little or no interaction takes place between the user and the system while a batch program is processing.

   (See System/38 Application Programming, Modules 2, 5, and 10.)

5. A qualified object name is the name of the object followed by the name of the library in which the object is stored. They are connected by a period; for example, OBJECT.LIBRARY.

   (See System/38 Control Program Facility Programmer's Guide chapter titled ''Objects'', Accessing Objects.)

6. A variable field is identified in a CL program with an & as the first character of the name. For example, the variable field DATE would be coded as &DATE.

   The field is defined in a DCL (Declare CL Variable) statement within the CL program or in a display file used by the CL program.

   (See System/38 Application Programming, Module 15.)

7. A library list is an ordered list of library names used to find an object. The library list indicates which libraries to search and the order in which they are to be searched for an object. If an object is in multiple libraries, you get the first one found, unless you use a qualified object name.

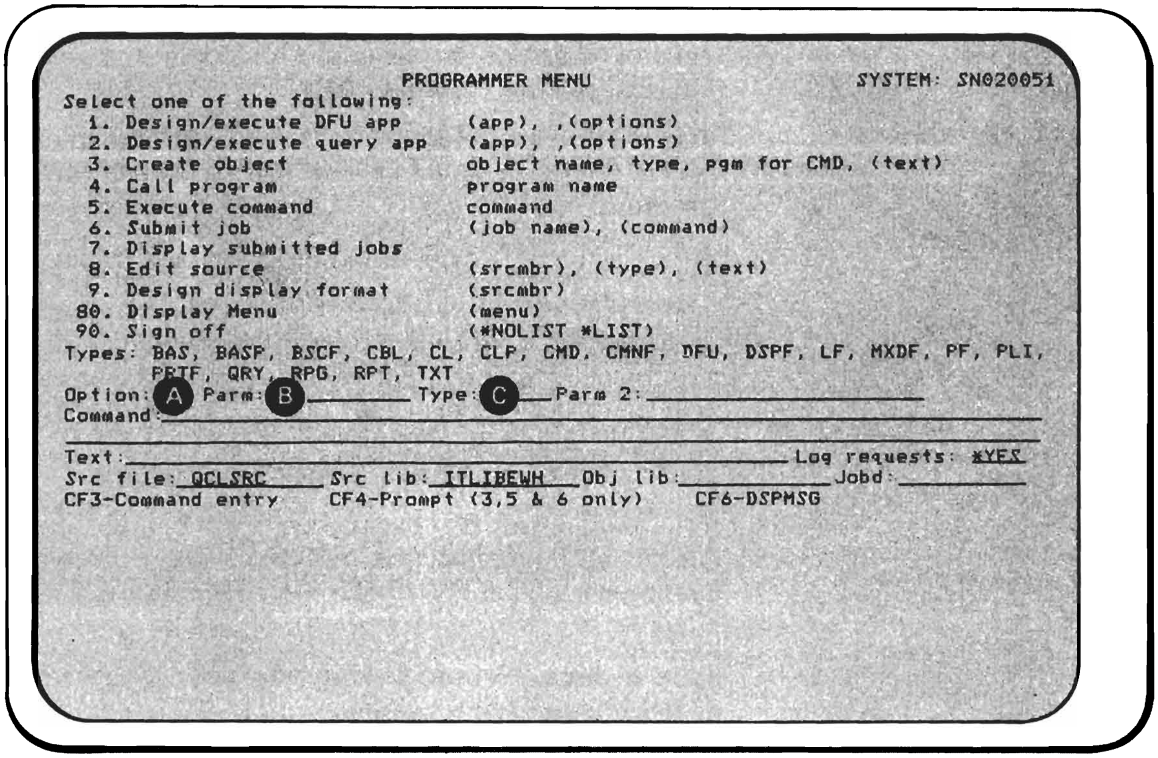
   A library list can be changed with the RPLLIBL (Replace Library List) command, the ADDLIBLE (Add Library List Entry) command, or the RMVLIBLE (Remove Library List Entry) command.

   (See System/38 Control Program Facility Programmer's Guide chapter titled ''Objects'', Using a Library List.)

Continue with Part II of the review on the next page.

Part II. Source Entry

8. Assume you wish to enter source code for a control language program named CLPROGA into the System/38. What entries would you make in the areas marked A, B, and C of the Programmer Menu shown below?



```
                          PROGRAMMER MENU                    SYSTEM: SN020051
Select one of the following:
  1. Design/execute DFU app      (app), ,(options)
  2. Design/execute query app    (app), ,(options)
  3. Create object               object name, type, pgm for CMD, (text)
  4. Call program                program name
  5. Execute command             command
  6. Submit job                  (job name), (command)
  7. Display submitted jobs
  8. Edit source                 (srcmbr), (type), (text)
  9. Design display format       (srcmbr)
 80. Display Menu                (menu)
 90. Sign off                    (*NOLIST *LIST)
Types: BAS, BASP, BSCF, CBL, CL, CLP, CMD, CMNF, DFU, DSPF, LF, MXDF, PF, PLI,
       PRTF, QRY, RPG, RPT, TXT
Option:   Parm:  __________Type:  ___Parm 2:______________________________
Command:__________________________________________________________________
__________________________________________________________________________
Text:_____________________________________________________Log requests: *YES
Src file: QCLSRC     Src lib: ITLIBEWH   Obj lib:______________Jobd:__________
CF3-Command entry   CF4-Prompt (3,5 & 6 only)    CF6-DSPMSG
```

A ____________________________________

B ____________________________________

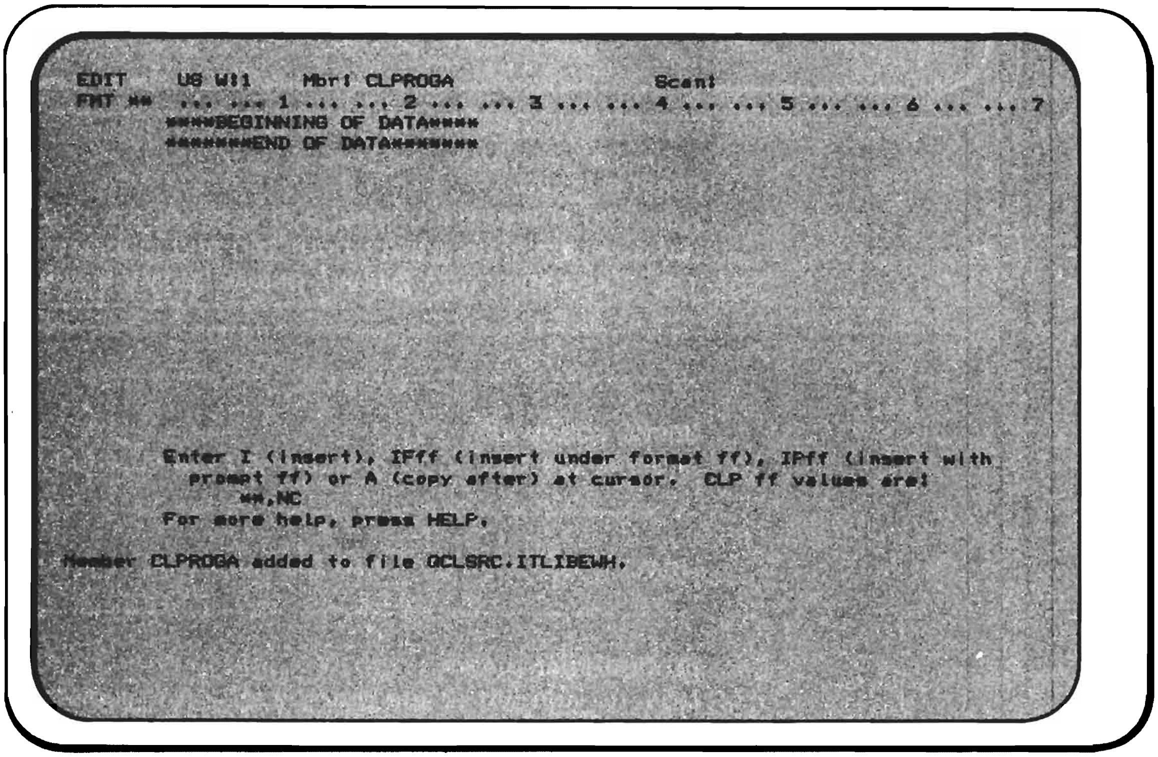
C ____________________________________

9. Referring to the Programmer Menu of the previous question, where will the source code for CLPROGA be stored in the System/38?

____

____

10. Based on the correct entries on the screen of Question 8, the screen below appears. How can you request prompting assistance for your input?



```
EDIT    US W:1    Mbr: CLPROGA                     Scan:
FMT **  ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
      ****BEGINNING OF DATA****
      ********END OF DATA********









      Enter I (insert), IFff (insert under format ff), IPff (insert with
        prompt ff) or A (copy after) at cursor.  CLP ff values are:
            **,NC
      For more help, press HELP.

Member CLPROGA added to file QCLSRC.ITLIBEWH.
```

11. You have finished entering the source code for CLPROGA. How do you tell SEU you are done?

Review your answers for Part II with those on the following pages.

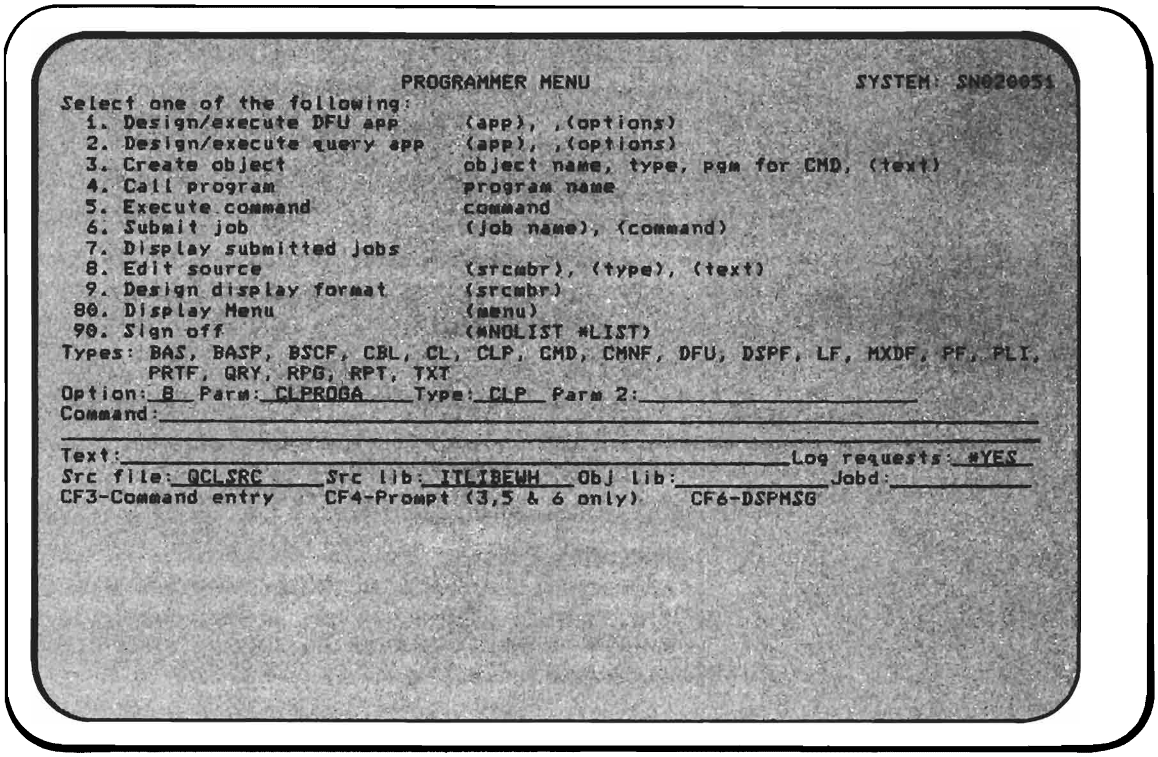
8. The three entries –Ⓐ,Ⓑ, and Ⓒ– identify the option you want from the Programmer Menu and parameters you wish to pass. The correct entries are:

   Ⓐ **8,** for menu option 8, Edit source

   Ⓑ **CLPROGA,** the source member name

   Ⓒ **CLP,** for control language program, the type of statements you want to enter

   The completed display is shown below.



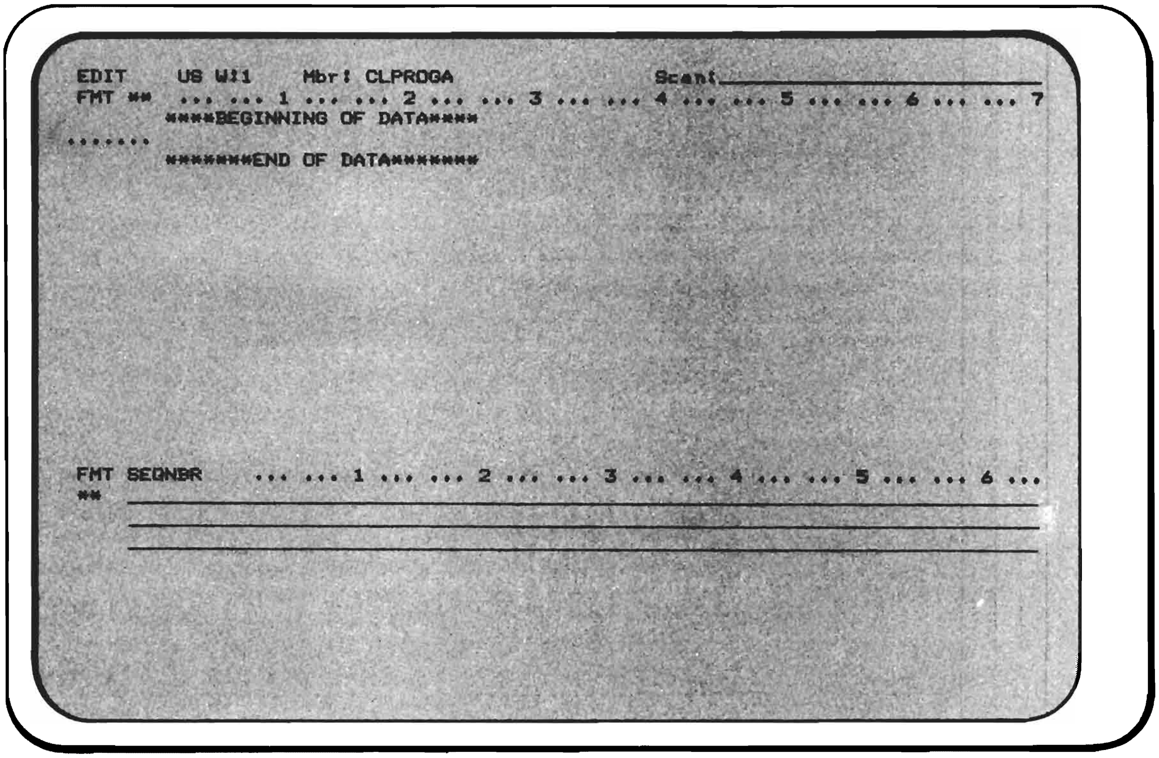
```
                            PROGRAMMER MENU                     SYSTEM: SN020051
Select one of the following:
  1. Design/execute DFU app        (app), ,(options)
  2. Design/execute query app      (app), ,(options)
  3. Create object                 object name, type, pgm for CMD, (text)
  4. Call program                  program name
  5. Execute command               command
  6. Submit job                    (job name), (command)
  7. Display submitted jobs
  8. Edit source                   (srcmbr), (type), (text)
  9. Design display format         (srcmbr)
 80. Display Menu                  (menu)
 90. Sign off                      (*NOLIST *LIST)
Types: BAS, BASP, BSCF, CBL, CL, CLP, CMD, CMNF, DFU, DSPF, LF, MXDF, PF, PLI,
       PRTF, QRY, RPG, RPT, TXT
Option:_8__Parm:_CLPROGA____Type:_CLP__Parm 2:_______________________
Command:_____________________________________________________________________
_____________________________________________________________________________
Text:_____________________________________________________Log requests:_*YES_
Src file:_QCLSRC_____Src lib:_ITLIBEWH___Obj lib:____________Jobd:___________
CF3-Command entry    CF4-Prompt (3,5 & 6 only)    CF6-DSPMSG
```

(See System/38 Application Programming, Module 3, Unit 3, The Programmer Menu.)

9. The source for CLPROGA will be stored in the source file named QCLSRC in the library named ITLIBEWH. These values come from the **Src file:** and **Src lib:** entries at the bottom of the Programmer's Menu.

   (See System/38 Application Programming, Module 3, Unit 3, The Programmer Menu.)

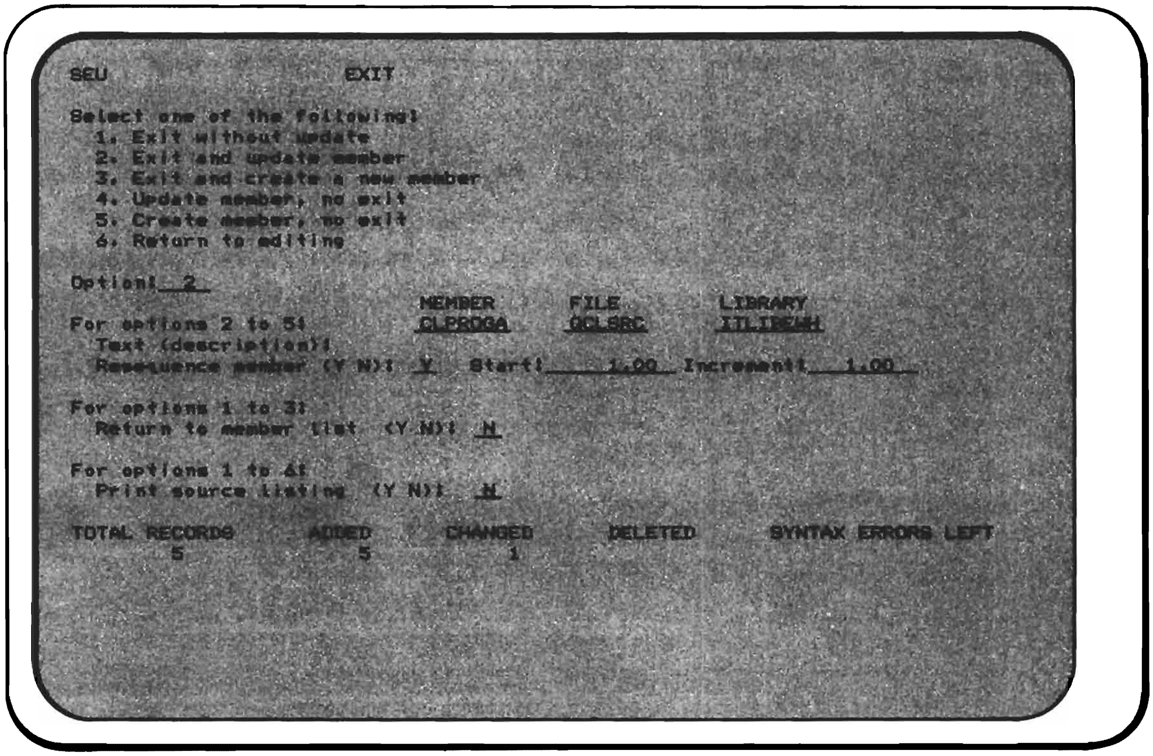
10. To request prompting assistance, key IP** (for **I**nsert with **P**rompting using format **) and press the Enter key. The display shown below appears on the screen. Notice the change at the bottom of the display. (Note: You could also enter IPNC, but your statements would not be syntax checked.)



```
EDIT    US W:1    Mbr: CLPROGA                      Scan:________________________
FMT **   ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
        ****BEGINNING OF DATA****
.......
        *******END OF DATA********









FMT SEQNBR     ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ...
**   ________________________________________________________________________
     ________________________________________________________________________
     ________________________________________________________________________
```

(See System/38 Application Programming, Module 4, Unit 4, Online Method (SEU); be sure to view the video presentation.)

11. To end SEU activity, press CF 1. A display like the one below appears. You can change the values shown on this display, return to SEU activity, or press the Enter key to end SEU.

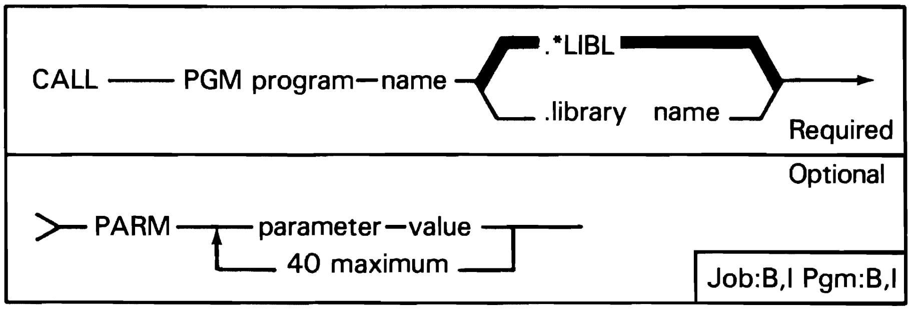


```
SEU                    EXIT

Select one of the following:
  1. Exit without update
  2. Exit and update member
  3. Exit and create a new member
  4. Update member, no exit
  5. Create member, no exit
  6. Return to editing

Option: _2_
                              MEMBER       FILE        LIBRARY
For options 2 to 5:           CLPROGA      QCLSRC      ITLIBEWH
  Text (description):
  Resequence member (Y N): Y  Start:     1.00 Increment:   1.00

For options 1 to 3:
  Return to member list  (Y N): N

For options 1 to 6:
  Print source listing  (Y N):  N

TOTAL RECORDS       ADDED       CHANGED       DELETED       SYNTAX ERRORS LEFT
      5                 5           1
```

Continue with Part III of the review on the next page.

Part III. Control Language Commands

Refer to the command syntax diagram shown below to answer Questions 12, 13, and 14.



12. What, if any, parameters are required with the CALL command?

13. Code the command to CALL the program AP009 located in the APLIB library. No parameters are passed to AP009.

14. Which, if any, of the commands below is in error and why?

```
a. CALL  PGM(PR001) PARM(&END)
b. CALL  PR002
c. CALL  PR003,PRLIB
d. CALL  PGM(PR004) +
         PARM(&CHECK)
e. CALL  PGM(PR005` PARM(&DATE)
```

15. Briefly describe what is being done with the command shown below.

```
CRTPF  FILE(APPTRN.GLC) SRCFILE(QDDSSRC.GLC)
```

16. Adding the following parameters to the CRTPF command above would have what affect on the physical file APPTRN?

```
MAINT(*REBLD)

SIZE(5000 500 10)
```

Review your answers with those on the next page.

Part III. Control Language Commands – Answers and Discussion

12. The CALL command has one required parameter – the program name (PGM).

    (See System/38 Application Programming, Module 15, or the Control Language Reference Manual chapter ''Format of Command Descriptions''.)

13. The command is coded:

    ```
    CALL PGM(AP009.APLIB)
    ```
    or
    ```
    CALL AP009.APLIB
    ```

    (See the Control Language Reference Manual chapter titled ''Control Language Syntax'' and the CALL command in the same manual.)

14. a. CALL PGM(PR001) PARM(&END) is a valid command

    b. CALL PR002 is a valid command

    c. CALL PR003,PRLIB is incorrect. If PRLIB is a library name used to create a qualified object name, a period is used to connect the values (PR003.PRLIB).

       If PRLIB is a parameter (PARM) value, a blank space should separate it from the PGM value.

    d. CALL PGM(PR004) + PARM(&CHECK) is a valid command. The + (plus sign) indicates the command is continued on the next line.

    e. CALL PGM(PR005 PARM(&DATE) is incorrect. The right parenthesis is missing from the PGM value. It should be PGM(PR005).

15. This command is creating a physical file, The file is named APPTRN and is stored in the library named GLC. The DDS source for this file is in a source file named QDDSSRC also in the library named GLC.

16. The parameter MAINT(*REBLD) causes the access path to be rebuilt each time the file is opened.

    The parameter SIZE(5000 500 10) sets the initial size of the file to 5000 records. It can be expanded in groups of 500 records 10 times.

    (For questions 15 and 16, see System/38 Application Programming, Module 4, Unit 4, File Creation Commands.)

Continue with Part IV of the review on the next page.

Part IV. Command Entry

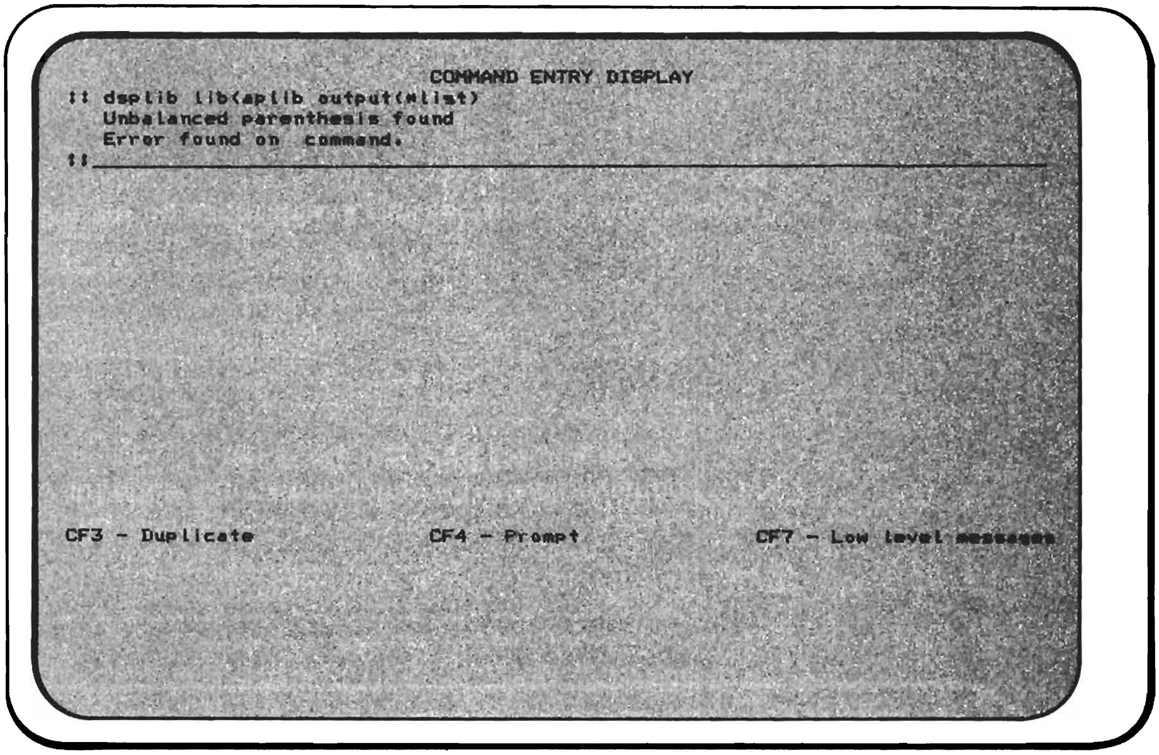
17. What is the primary function of the Command Entry Display?

18. How do you get to the Command Entry Display from the Programmer Menu?

    How do you then return to the Programmer Menu?

19. What is the use of CF 4 when you are entering a control language command?

20. The Command Entry Display below shows a command entered with a syntax error detected and displayed by CPF. How can you easily correct and re-enter the command by using a CF key?



```
                              COMMAND ENTRY DISPLAY
:: dsplib lib(aplib output(*list)
   Unbalanced parenthesis found
   Error found on  command.
::______________________________________________________________________________




CF3 - Duplicate                  CF4 - Prompt                CF7 - Low level messages
```

Review your answers with those on the following pages.

17. The primary function of the Command Entry Display is to allow you to enter control language commands.

18. To get to the Command Entry Display from the Programmer Menu, press CF 3.

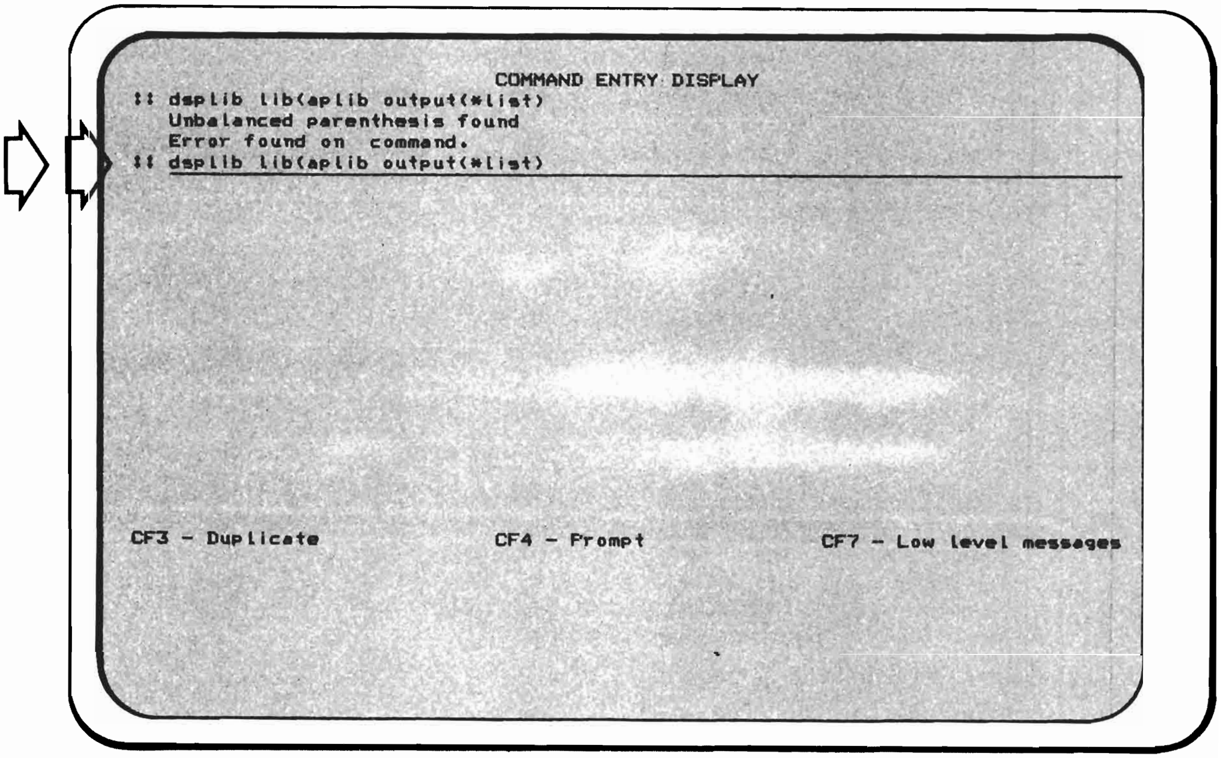
    To get back to the Programmer Menu from the Command Entry Display, press CF 1.

19. CF 4 allows you to be prompted for the parameters of a command.

    (See System/38 Application Programming, Module 3, Unit 2 for a review of questions 17, 18, and 19.)

20. To correct (or re-use) a command already entered on the Command Entry Display, first move the cursor to the line containing the command you want.

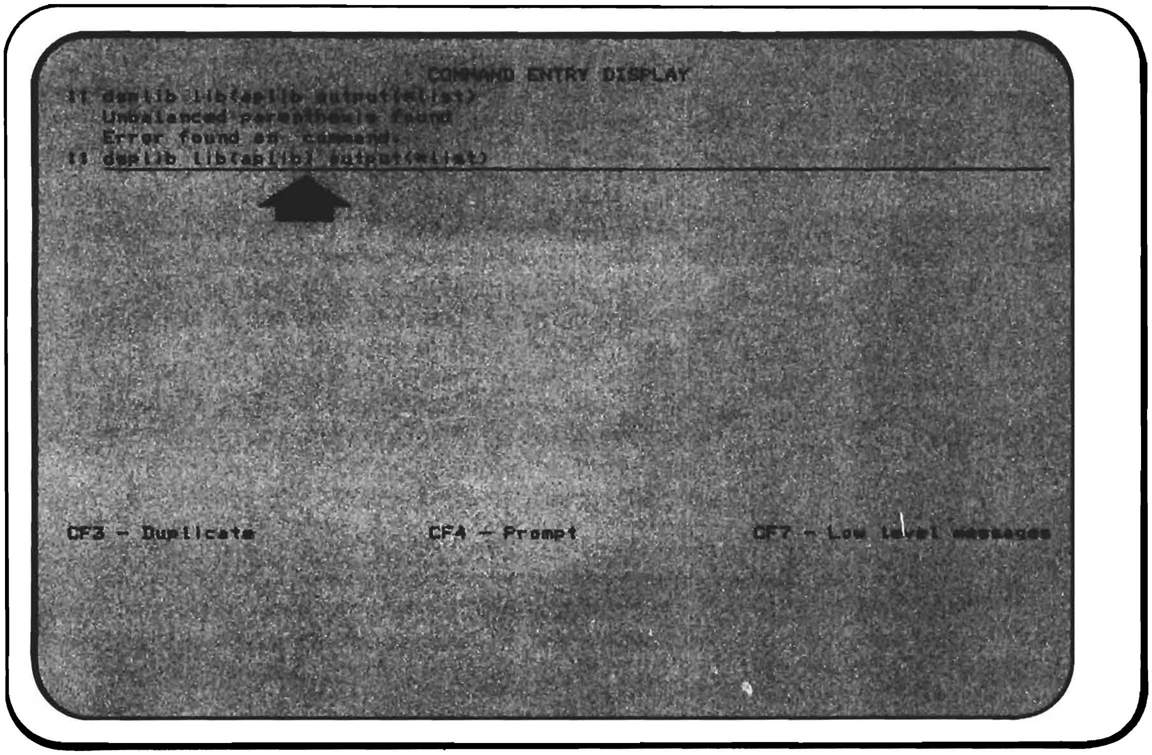
    Next, press CF 3 to duplicate the command on the next available line on the Command Entry Display, as shown on the display below.



```
                          COMMAND ENTRY DISPLAY
:: dsplib lib(aplib output(*list)
   Unbalanced parenthesis found
   Error found on  command.
:: dsplib lib(aplib output(*list)




CF3 - Duplicate             CF4 - Prompt             CF7 - Low level messages
```

(Answer continued on next page.)

Position the cursor and correct the error. In the command used for the question, the error is an omitted parenthesis. To enter it, position the cursor, press the Insert key, and type the) you need. The display below shows the corrected command.



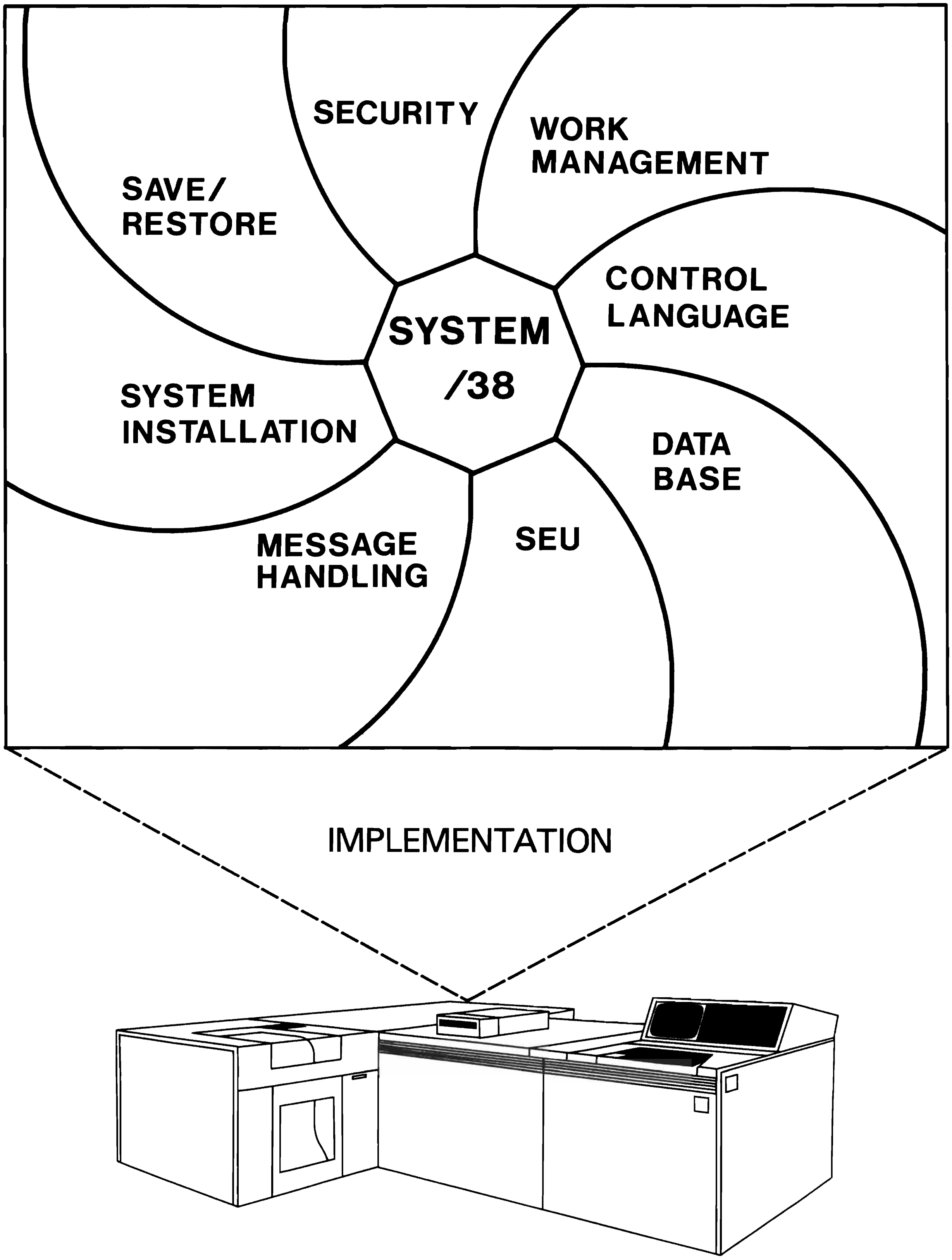
```
                          COMMAND ENTRY DISPLAY
:: dsplib lib(aplib output(*list)
   Unbalanced parenthesis found
   Error found on command.
:: dsplib lib(aplib) output(*list)







CF3 - Duplicate              CF4 - Prompt              CF7 - Low level messages
```

Press Enter Rec Adv to submit the command to CPF for processing.

(See System/38 Application Programming, Module 3, Unit 2, The Command Entry Display.)

This concludes the review section. If you were unsure about any of the answers, you should study the references indicated with those answers before continuing with System/38 Implementation Topics.

When you feel you are ready, continue with Section III of this Student Materials Book.


SECURITY
WORK MANAGEMENT
SAVE/ RESTORE
CONTROL LANGUAGE
SYSTEM /38
SYSTEM INSTALLATION
DATA BASE
SEU
MESSAGE HANDLING
IMPLEMENTATION

Section III

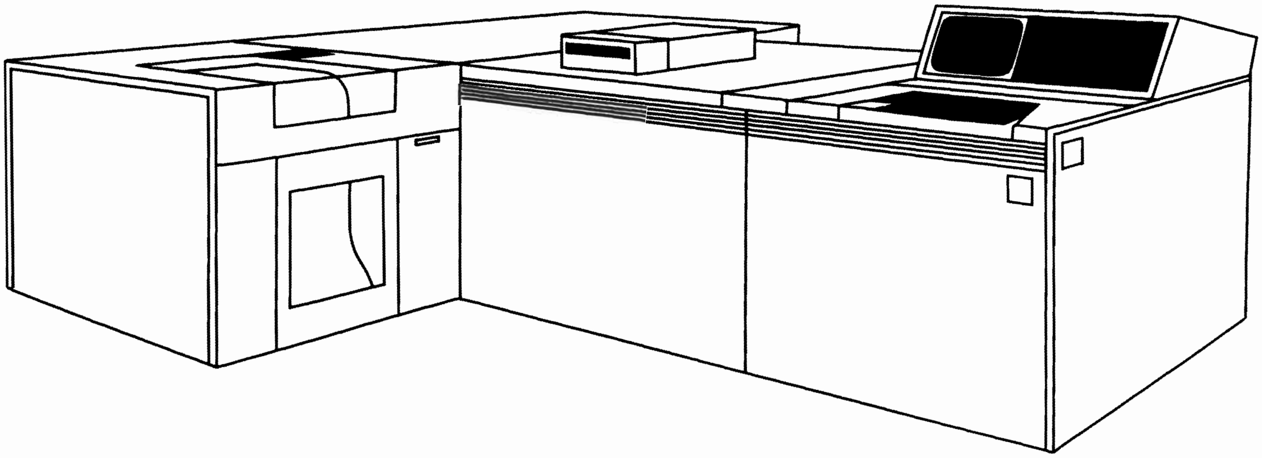
Implementation Topics Notes

The remainder of this book is divided into eight parts, one for each module of the course. Within each part you find a brief introduction to the module, a place for your personal notes, machine exercise, progress checks, and a summary of the material covered in the module. In addition, some parts have illustrations, diagrams, and sample forms to aid your implementation effort.

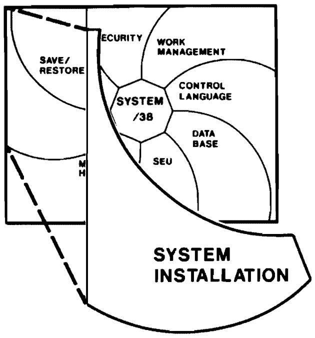
- Introduction . . . provides you with an overview of the module and lists any reference manuals you may need. Always begin your study of a module with the Introduction, here, in your Student Materials Book.
- Personal Notes . . . as you study the various modules in the course, certain topics, techniques, or references may be especially relevant to your needs. Use the personal notes to record these thoughts while they are fresh in your mind. This note space should be used to write your answers to any desk exercises that a module may have. You will have a permanent record of your activity if you write all your notes in the same place, your Student Materials Book.
- Machine Exercises . . . to reinforce the material covered. They are a means for you to practice what you have studied in the text. Directions in the module text tell you when to do a specific machine exercise.
- Progress Checks . . . to also reinforce the material covered in a study unit. Use it as a review of the information you have learned. They are a means for you to see if you understand what you have just studied.
- Summary . . . to help you put the module into perspective. Notes, diagrams, and comments about the topic's role in the implementation effort are included, as appropriate. Do not hesitate to add your own thoughts to these notes.

The text, your personal notes, your Progress Check answers and your copies of the System/38 reference manuals become the foundation for the implementation of your system. You will use them to guide your activities in the installation of new applications on your System/38.

You should now be ready to study the first topic, ''System Installation''. The introduction to this topic starts on the next page.



Module 1. System Installation



The responsibility of installing your system may lie with one person or several people in your company. You need to know how to install program products and configure devices on your System/38. Module 1 shows you how to prepare for device configuration, how to install the Control Program Facility (CPF) and other program products, and how to use control language commands to configure your devices.

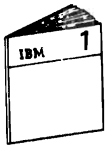
You may wish to modify your system in accordance with Module 1 recommendations or you may wish to use your system as you receive it from IBM. Module 1 also shows you how to save your system once it is installed.

You should plan for device configuration before your system arrives. You need to fill out the necessary work sheets provided and keep them for ready reference and updating whenever you add, remove, or relocate devices on your system.

When your system arrives, use the procedures covered in Module 1 to install CPF, your program products, and to configure your devices. To ensure your installation proceeds as smoothly as possible, read the Guide To Program Product Installation and Device Configuration (GC21-7775) **before** you actually perform the installation activities.

You will use the following materials as you study this module:

- The Module 1 text
- IBM System/38 Guide To Program Product Installation and Device Configuration (GC21-7775)
- IBM 5250 Information Display System Planning and Site Preparation Guide (GA21-9337)
- IBM System/38 Control Language Reference Manual (SC21-7731)
- IBM System/38 Control Program Facility Programmer's Guide (SC21-7730)
- IBM 5292 Color Display Station Setup Procedure (GA21-9415)



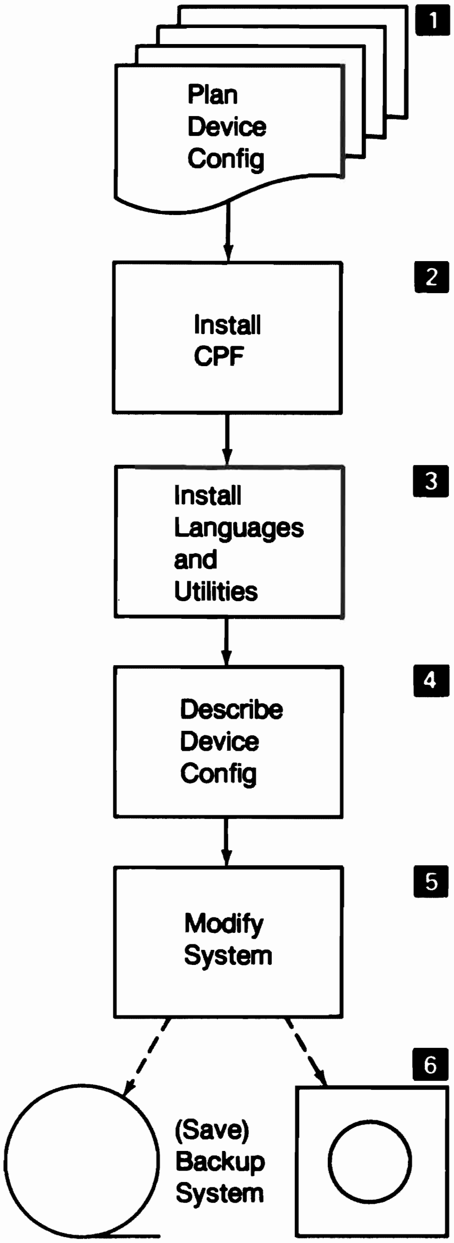
Now begin your study of System Installation in the Module 1 text.

STUDENT NOTES: Module 1. System Installation

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

STUDENT NOTES (continued): System Installation

SYSTEM INSTALLATION STEPS



1 Plan for device configuration – this consists of filling out the work sheets for your system. You should complete this step before your system arrives.

2 Install your system, including the Control Program Facility (CPF) and the general-purpose library (QGPL).

3 Install and verify your languages and utilities.

4 Describe your configuration (local and remote).

5 Tailor your system.

6 Save your system.

SYSTEM PRINTER WORK SHEET

SYSTEM PRINTER
(CRTDEVD command)

Description		**Parameter**	**Entry**
Name of the system printer.	R	DEVD	____________
Physical address of the device:	R	DEVADR	________

Device	**Entry**
First system printer	
3262 or 5211	000018
3203 or 4245	000040
Second system printer	
3262 or 5211	000058
3203 or 4245	000040 If first system printer is a 3262 or 5211.
3203 or 4245	000041 If first system printer is a 3203 or 4245.

Device type (3262, 5211, 3203, or 4245).	R	DEVTYPE	________
Device model.	R	MODEL	________

Device Type	**Model**	**Entry**
3262	A1	A1
	B1	B1
5211	2	2
3203	5	5
4245	12	12
	20	20

The device is to be varied online when CPF is started (*NO or *YES).	ONLINE	________
The name of the default print image. (IBM-supplied print image is QSYSIMAGE in QGPL.)	PRTIMG	____________ __________
The authority for this device to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	________
Brief description of the device. (*BLANK or no more than 50 characters in apostrophes.)	TEXT	____________

____

____

When you are ready, return to the module text to check your entries on your work sheet for the first system printer.

LOCAL WORK STATION CONTROLLERS

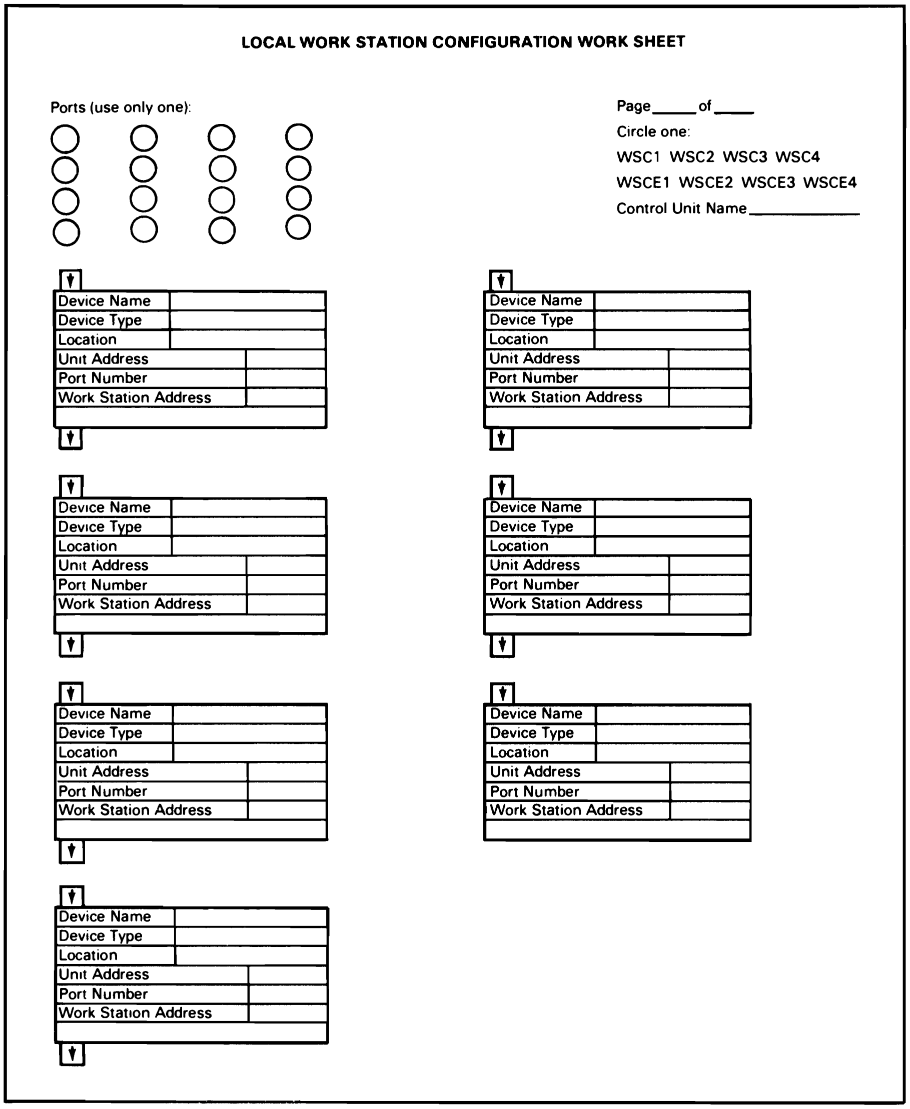
FEATURE AVAILABLE	MAX. NO. OF WORK STATIONS	MAX. NO. OF PORTS
Work Station Controller (WSC) (without either expansion feature)	12	8
Work Station Controller (WSC with Device Control Expansion feature)	20	8
Work Station Controller (WSC with Device Interface Expansion feature)	20	16
Work Station Controller-Extended (WSCE)	32	8

Please keep in mind the following guidelines as you configure your work stations:

- A maximum of seven work stations can be attached to a single port.
- The number of work stations that can be attached to a work station controller depends on the type of controller.
- The WSCE does not support the 5251 Model 1 and the 5252 Displays.

Desk Exercise – Unit 1: Pre-installation

Using the work sheet below, fill out the work sheet for six 5251 Model 11 display stations and one 5224 Model 1 work station printer. These devices are to be attached to the third port, which is port number 02, on Controller 1 (WSC1). All stations except the printer have the Cable-Thru feature.



LOCAL WORK STATION CONFIGURATION WORK SHEET

Ports (use only one):

○ ○ ○ ○
○ ○ ○ ○
○ ○ ○ ○
○ ○ ○ ○

Page_____of_____
Circle one:
WSC1 WSC2 WSC3 WSC4
WSCE1 WSCE2 WSCE3 WSCE4
Control Unit Name___________

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

After you complete this exercise, return to the Module text to check your work.

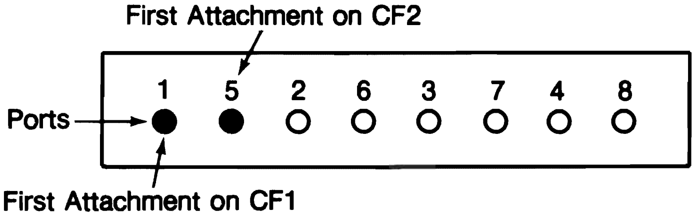
5291

5251 REMOTE CONTROLLER – WORK STATION ATTACHMENT GUIDELINES

1. The maximum number of work stations that can be attached to each port of a cluster feature is four (using the Cable Thru feature).

2. The maximum number of work stations that can be attached to a cluster feature is four (4 ports per cluster feature). Overall, the maximum number of work stations that can be attached to the 5251 controller is 8 (with the Dual Cluster feature).

3. Ports must be used in sequence; CF1 (1 to 4), CF2 (5 to 8). See diagram.



4. One work station on each port used must have a work station address of 0. Other work station addresses must be 1, 2, or 3.

5. A work station with an address of 0 must be the last station on the port.

6. The Cluster feature you select must match one of the configurations shown in the cluster configuration charts in the "Remote Work Station Configuration Using the 5251" chapter of the IBM 5250 Information Display System Planning and Site Preparation Guide (GA21-9337). Please review these charts at this time.

Progress Check – Unit 1: Pre-installation

Read each question carefully. Record your answers in the space provided.

1. List the 6 steps to install a System/38.

   A. Plan for device configuration
   B Install CPF
   C. Install languages + utilities
   D Describe your device configuration
   E Tailor your system
   F Save your system

2. Must all devices be described to the System/38? If yes, how?

   Yes. Some system device descriptions are supplied with the system. You describe other devices with the Create Device Description (CRTDEVD) command.

3. What is the purpose of filling out configuration and device work sheets before system installation?

   The purpose of completing the work sheets before system installation is to enable you to enter commands quickly and easily when you actually configure your system.

4. Do you have to create device descriptions for the console and diskette system devices?

   No. These device descriptions are shipped with your system.

5. What feature is required on the work station to attach another work station?

The Cable-Thru feature. This feature permits the attachment of more than one work station to a single port.

6. How many local work stations can be attached (using the Cable-Thru feature) to one port on the work station controller?

You may attach a maximum of seven work stations to a single port on the work station controller.

Skip questions 7 and 8 if you did not study the section on Remote Work Stations in the module text.

7. How many remote work stations can be attached to a cluster feature on the 5251 Model 12 controller?

The maximum number of work stations that can be attached to a cluster is four.

8. What are the 5251 remote controller switches and the function of each?

A. Controller Station Address Switch - represents the controller station address.

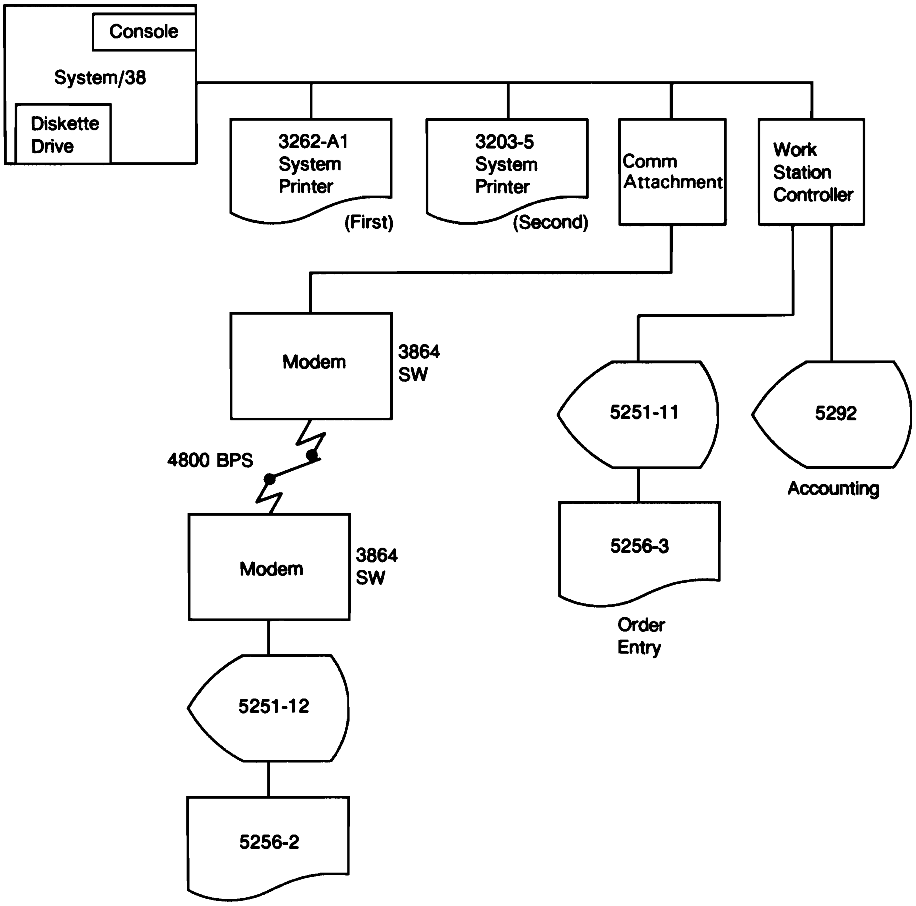
B. Communications Line Configuration Switch - represents the type of communications configuration used.

C. Cluster Feature Port Switch - determines which port the 5251 controller searches for work stations.

When you are finished, return to the module and review your answers with those in the text.

Sample Completed Work Sheets

The following pages contain the completed work sheets for the sample system you have seen in the module text. Use them for reference and as a guide to completing the necessary work sheets for your system.



LOCAL WORK STATION CONFIGURATION WORK SHEET

Ports (use only one):

ØØ	○	○	○
○	○	○	○
○	○	○	○
○	○	○	○

Page 1 of 2

Circle one:

[✓ WSC1] WSC2 WSC3 WSC4

WSCE1 WSCE2 WSCE3 WSCE4

Control Unit Name QWSC1

Device Name	WSØØ1
Device Type	5251-11
Location	ORDER ENTRY
Unit Address	ØØ
Port Number	ØØ
Work Station Address	ØØ

Device Name	WSPR1
Device Type	5256-3
Location	ORDER ENTRY
Unit Address	Ø1
Port Number	ØØ
Work Station Address	Ø1

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

LOCAL WORK STATION CONFIGURATION WORK SHEET

Page 2 of 2

Circle one:

[✓ WSC1] WSC2 WSC3 WSC4

WSCE1 WSCE2 WSCE3 WSCE4

Control Unit Name QWSC1

Ports (use only one):

○ [✓ Ø1] ○ ○
○ ○ ○ ○
○ ○ ○ ○
○ ○ ○ ○

Device Name	WSØØ2
Device Type	5292-1
Location	ACCOUNTING
Unit Address	Ø2
Port Number	Ø1
Work Station Address	ØØ

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

Device Name	
Device Type	
Location	
Unit Address	
Port Number	
Work Station Address	

LOCAL WORK STATION CONTROLLER
(CRTCUD command)

Description		Parameter	Entry
Name of the control unit.	R	CUD	QWSC1
Control unit type identifier (*WSC or *WSCE).	R	TYPE	*WSC
Model number of the control unit (*NONE):	R	MODEL	*NONE
Address of the control unit:	R	CTLADR	ØØ3Ø

Type	Entry
WSC1 or WSCE1	0030
WSC2 or WSCE2	0070
WSC3 or WSCE3	00B0
WSC4 or WSCE4	00F0

The control unit is to be varied online when CPF is started (*YES or *NO).		ONLINE	*YES
List *on this work sheet only* (not on the CRTCUD command prompt itself) the name(s) of the devices to be attached to this control unit (up to 20 on WSC; up to 32 on WSCE). *Do not enter values for the DEV parameter on the CRTCUD command prompt.* When you create individual device descriptions for display devices and work station printers, and you reference this control unit through the CTLU parameter, those device names are automatically inserted in the DEV parameter for this control unit. (See the appropriate *Local Work Station Configuration Work Sheet.*) (Use additional sheets if necessary.)		DEV	
The authority for this control unit to be granted to all users (*NORMAL, *ALL, or *NONE).		PUBAUT	*NORMAL
Brief description of the control unit. (*BLANK or no more than 50 characters, enclosed in apostrophes.)		TEXT	

'Work Station Controller 1'

Work Sheet for a Local Work Station Controller (WSC or WSCE)

DISPLAY STATION (PART 1 OF 2)
(CRTDEVD command)

Description		Parameter	Entry
Name of the display station. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)	R	DEVD	WSØØ1
Physical address of the device:	R	DEVADR	ØØØØØØ

Control Unit	Entry
WSC or WSCE	000000
5251	xxyyyy

xxyyyy:

- yyyy — CTLADR parameter values from CRTCUD work sheet
- xx — Unit address (00 if device is part of 5251 Model 2 or 12; 02-05 if attached to first cluster; 06-09 if attached to second cluster)

Description		Parameter	Entry
Device type (5251, 5252, 5291, or 5292).	R	DEVTYPE	5251
Device model:	R	MODEL	11

Device Type	Screen Size	Entry
5251	960 characters	1
	1920 characters	11
5252	960 characters (dual)	1
5291	1920 characters	1
5292	1920 characters	1

Description		Parameter	Entry
Name of associated work station controller or 5251 control unit. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)		CTLU	QWSC1
This device is varied online when CPF is started (*NO or *YES).		ONLINE	*YES

Work Sheet for a Display Station

DISPLAY STATION (PART 2 OF 2)
(CRTDEVD command)

Description	Parameter	Entry
The line connection (switched lines only) is to be broken after the work station user signs off without specifying a value for the DROP parameter on the SIGNOFF command (*NO or *YES).	DROP	________
Name of the associated work station printer (*NONE or device name).	PRINTER	WSPR1
Name of an alternative printer file to be used when no associated work station printer is available.	PRTFILE	________
Address of device:	WSCADR	ØØØØØØ

Control Unit	Entry
5251	000000
WSC or WSCE	xxyyzz

```
xxyyzz
│ │ └──── Work station address switch settings (00-06)
│ │
│ └────── Work station controller port number as follows:
│
│                    Valid             Valid
│          WSC       Entries   WSCE    Entries
│          WSC1      00-15     WSCE1   00-07
│          WSC2      16-31     WSCE2   16-23
│          WSC3      32-47     WSCE3   32-39
│          WSC4      48-63     WSCE4   48-55
└──────── Unit address (00-19 if WSC; 00-31 if WSCE)
```

(See the appropriate *Local Work Station Configuration Work Sheet.*)

Description	Parameter	Entry
Type of keyboard (only when display station is connected to WSC or WSCE):	WSCKBD	TUSB

Entry

```
yzzz
│└────────── 3-character identifier (see CRTDEVD command in CL Reference
│            Manual)
│           ┌ T for typewriter-like keyboard
└──────────{  D for data entry keyboard without proof arrangement
            └ P for data entry keyboard with proof arrangement
```

Description	Parameter	Entry
Application program is to control blinking cursor (*YES or *NO).	ALWBLN	________
The authority for this device to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	*NORMAL
Brief description of the device. (*BLANK or no more than 50 characters, enclosed in apostrophes.)	TEXT	

'Order Entry Department Display Station'

____

Work Sheet for a Display Station

WORK STATION PRINTER (PART 1 OF 2)
(CRTDEVD command)

Description		Parameter	Entry
Name of the work station printer. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)	R	DEVD	WSPR1
Physical address of the device:	R	DEVADR	ØØØØØØ

Control Unit	Entry
WSC or WSCE	000000
5251	xxyyyy

yyyy — CTLADR parameter value from CRTCUD work sheet

xx — Unit address (02-05 if attached to first cluster; 06-09 if attached to second cluster)

(See the appropriate *5250 Communications Network Setup Form.*)

Device type (5219, 5224, 5225, or 5256).	R	DEVTYPE	5256
Device model:	R	MODEL	3

Device Type	Model	Entry
5219	D1 D2	D1 D2
5224	1	1
	2	2
5225	1	1
	2	2
	3	3
	4	4
5256	1 2 3	1 2 3

Name of the associated work station controller or 5251 control unit. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)		CTLU	QWSC1
The device is to be varied online when CPF is started (*NO or *YES).		ONLINE	*YES

Work Sheet for a Work Station Printer

WORK STATION PRINTER (PART 2 OF 2)
(CRTDEVD command)

Description — **Parameter** — **Entry**

Name of the message queue to which operational messages should be sent. — MSGQ — WSØØ1

Address of device: — WSCADR — Ø1ØØØ1

Control Unit	**Entry**
5251	000000
WSC or WSCE	xxyyzz

- zz — Work station address switch settings (00-06)
- yy — Work station controller port number as follows:

WSC	**Valid Entries**	**WSCE**	**Valid Entries**
WSC1	00-15	WSCE1	00-07
WSC2	16-31	WSCE2	16-23
WSC3	32-47	WSCE3	32-39
WSC4	48-63	WSCE4	48-55

- xx — Unit address (00-19 if WSC; 00-31 if WSCE)

(See the appropriate *Local Work Station Configuration Work Sheet.*)

The default font identifier (3 digits; any combination of 0-9) to be used if FONT is not specified for a printer file. Required for DEVTYPE (5219; valid only for DEVTYPE(5219)). — FONT — ______

The mode in which paper is to be fed to the printer (*CONT, *CUT, or *AUTOCUT). Valid only for DEVTYPE (5219). — FORMFEED — ______

The authority for this device to be granted to all users (*NORMAL, *ALL, or *NONE). — PUBAUT — *NORMAL

Brief description of the device. (*BLANK or no more than 50 characters, enclosed in apostrophes.) — TEXT

'Order Entry Department Work Station Printer'

Work Sheet for a Work Station Printer

DISPLAY STATION (PART 1 OF 2)
(CRTDEVD command)

Description		Parameter	Entry
Name of the display station. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)	R	DEVD	WSØØ2
Physical address of the device:	R	DEVADR	ØØØØØØ

Control Unit	Entry
WSC or WSCE	000000
5251	xxyyyy

xx: Unit address (00 if device is part of 5251 Model 2 or 12; 02-05 if attached to first cluster; 06-09 if attached to second cluster)

yyyy: CTLADR parameter values from CRTCUD work sheet

Description		Parameter	Entry
Device type (5251, 5252, 5291, or 5292).	R	DEVTYPE	5292
Device model:	R	MODEL	1

Device Type	Screen Size	Entry
5251	960 characters	1
	1920 characters	11
5252	960 characters (dual)	1
5291	1920 characters	1
5292	1920 characters	1

Description		Parameter	Entry
Name of associated work station controller or 5251 control unit. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)		CTLU	QWSC1
This device is varied online when CPF is started (*NO or *YES).		ONLINE	*YES

Work Sheet for a Display Station

DISPLAY STATION (PART 2 OF 2)
(CRTDEVD command)

Description	Parameter	Entry
The line connection (switched lines only) is to be broken after the work station user signs off without specifyihg a value for the DROP parameter on the SIGNOFF command (*NO or *YES).	DROP	________
Name of the associated work station printer (*NONE or device name).	PRINTER	________
Name of an alternative printer file to be used when no associated work station printer is available.	PRTFILE	________
Address of device:	WSCADR	Ø2Ø1ØØ

Control Unit	Entry
5251	000000
WSC or WSCE	xxyyzz

xxyyzz
- zz — Work station address switch settings (00-06)
- yy — Work station controller port number as follows:

WSC	Valid Entries	WSCE	Valid Entries
WSC1	00-15	WSCE1	00-07
WSC2	16-31	WSCE2	16-23
WSC3	32-47	WSCE3	32-39
WSC4	48-63	WSCE4	48-55

- xx — Unit address (00-19 if WSC; 00-31 if WSCE)

(See the appropriate *Local Work Station Configuration Work Sheet.*)

Description	Parameter	Entry
Type of keyboard (only when display station is connected to WSC or WSCE):	WSCKBD	TUSB

Entry
yzzz
- zzz — 3-character identifier (see CRTDEVD command in *CL Reference Manual*)
- y — T for typewriter-like keyboard
D for data entry keyboard without proof arrangement
P for data entry keyboard with proof arrangement

Description	Parameter	Entry
Application program is to control blinking cursor (*YES or *NO).	ALWBLN	________
The authority for this device to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	*NORMAL
Brief description of the device. (*BLANK or no more than 50 characters, enclosed in apostrophes.)	TEXT	

'Accounting Department Display Station'

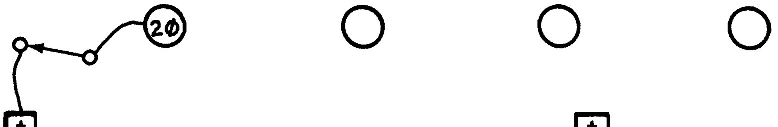
Work Sheet for a Display Station

REMOTE WORK STATION CONFIGURATION WORK SHEET

Communications
attachment (circle one): [✓ 1] 2 3

Page 1 of ____

Line Description
Name: LINE 2Ø ____________ ____________ ____________



Control Unit Name	RCU
Control Unit Type	5251-12
Control Unit Address	Ø1ØØ
Telephone	404-238-9999
Display Device Name	RWSØ1
Display Device Type	5251-11
Unit Address	ØØ
Location	Branch Office

Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

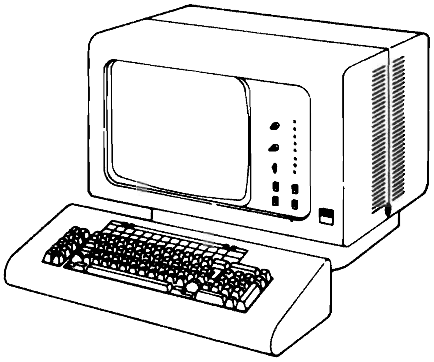
Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

Control Unit Name	
Control Unit Type	
Control Unit Address	
Telephone	
Display Device Name	
Display Device Type	
Unit Address	
Location	

IBM 5250 COMMUNICATIONS NETWORK SETUP FORM (Part 1)



5251 MODEL 12 DISPLAY STATION

5251 Model 12 Information

Name RCU

Location Branch Office

City, State Atlanta, Georgia

Telephone 4Ø4-238-9999

Host System Line/Port Number 2Ø

Location Main Office

Telephone 4Ø4-238-ØØØØ

Device Type 5251-12

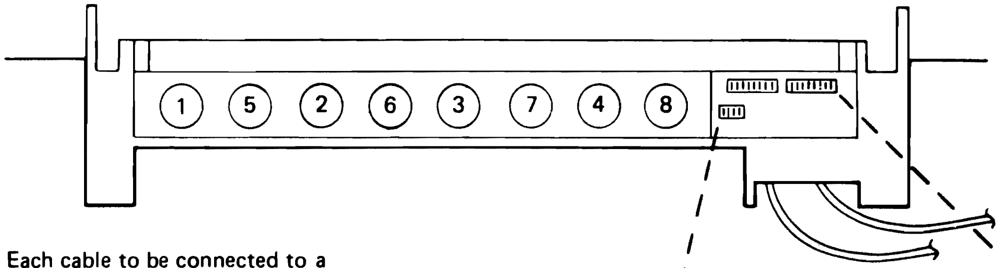
Controller Station Address Ø1

Unit Address 00

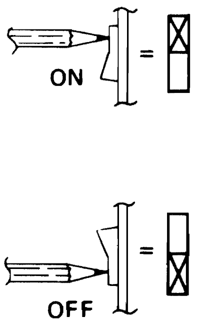
Work Station Address 0

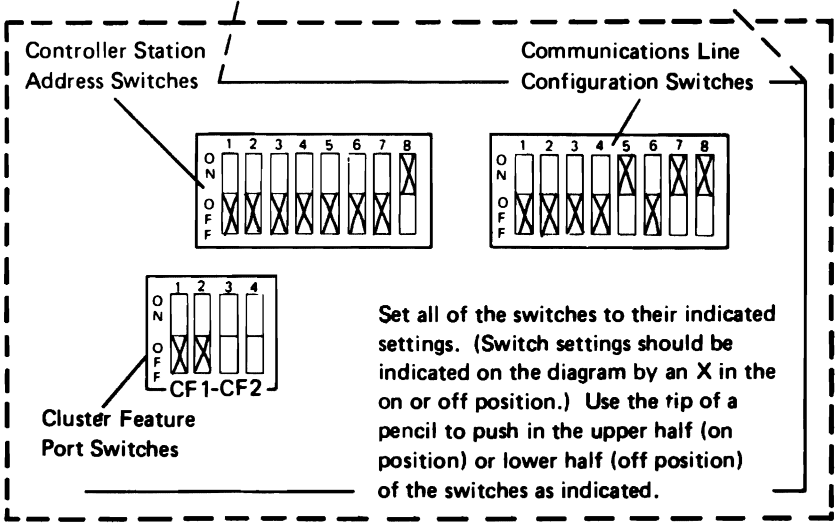
Communications Type EIA

CE assistance required for communications line connection? □ Yes □ No



Each cable to be connected to a 5251 Model 2 or 12 port should have a tag with a number from 1 through 8. There should be a cable for each port used as indicated on Part 2 of this form. Connect each cable to the port indicated on its tag.

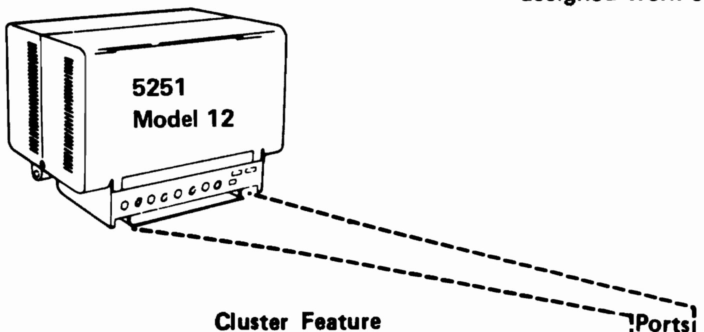



Controller Station Address Switches
Communications Line Configuration Switches
1 2 3 4 5 6 7 8
ON
OFF
1 2 3 4 5 6 7 8
ON
OFF
1 2 3 4
ON
OFF
CF1-CF2
Cluster Feature Port Switches

Set all of the switches to their indicated settings. (Switch settings should be indicated on the diagram by an X in the on or off position.) Use the tip of a pencil to push in the upper half (on position) or lower half (off position) of the switches as indicated.

IBM 5250 COMMUNICATIONS NETWORK SETUP FORM (Part 2)

Note: Set the address on your work station to your assigned work station address as shown in each box.



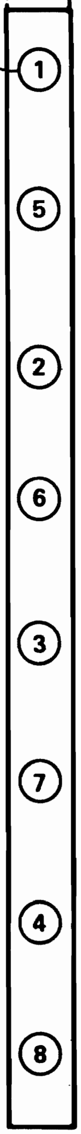
Cluster Feature

Name	RWSPR1
Device Type	5256-2
Location	Branch
Work Station Address	ØØ
Unit Address	Ø2
Telephone	

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	



With Dual Cluster Feature

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	

Name	
Device Type	
Location	
Work Station Address	
Unit Address	
Telephone	

Remote Work Stations: IBM 5250 Communications Network Setup Form

SDLC PRIMARY LINE (PART 1 OF 2)
(CRTLIND command)

Description		Parameter	Entry
Name of the line.	R	LIND	LINE 2Ø
Number that identifies the line:	R	LINNBR	2Ø

Line Position	Entry	Line Position	Entry	Line Position	Entry
First	20	Fifth	60	Ninth	A0
Second	21	Sixth	61	Tenth	A1
Third	22	Seventh	62	Eleventh	A2
Fourth	23	Eighth	63	Twelfth	A3

Type of line (*SDLCP).	R	TYPE	*SDLCP
Type of line connection:	R	CNN	*SWT

Connection Type	Entry
Switched	*SWT
Nonswitched point-to-point	*PP
Nonswitched multipoint	*MP

The line rate in bits per second (1200, 2000, 2400, 4800, 7200, 9600, 48000, or 56000).	R	RATE	48ØØ
The modem has the switched network (dial) backup feature (*NO or *YES). Not valid for CNN(*SWT).		SWNBKU	*NO
The modem has the data rate select feature (*NO or *YES).		SELECT	*YES
Nonreturn to zero inverted transmission decoding method is required (*NO or *YES).		NONRTNZ	*YES
System/38 provides clocking function for the line (*NO or *YES).		CLOCK	*NO
Autocall feature is installed (*NO or *YES). *YES is valid only with CNN(*SWT).		AUTOCALL	*NO
Autoanswer feature is installed (*NO or *YES). *YES is valid only with CNN(*SWT).		AUTOANS	*YES
System/38 provides answer tone signal to the modem (*NO or *YES). *YES is valid only with CNN(*SWT).		ANSTONE	*NO
The physical connection is by 2-wire or 4-wire link (2 or 4).		WIRE:	
		Normal:	2
		Backup:	
Data communications equipment group (*A, *B, or *C).		DCEGRP	*C
Non-IBM modem is used (*NO or *YES).		OEMMDM	*NO
Types of calls for which the line is to be used:		SWTCNN	*BOTH

Type	Entry
Both incoming and outgoing calls	*BOTH
Incoming calls only	*ANS
Outgoing calls only	*CALL

The speed at which the line operates (*FULL or *HALF).		RATETYPE	*FULL
Line connection is dialed manually (*MANUAL) or automatically (*AUTO). Valid only for CNN(*SWT).		DIALMODE	*MANUAL
Incoming calls are answered manually (*MANUAL) or automatically (*AUTO). Valid only for CNN(*SWT).		ANSMODE	*AUTO

SDLC PRIMARY LINE (PART 2 OF 2)
(CRTLIND command)

Description	Parameter	Entry
Number of delay time units (200 milliseconds each) before the system ends the operation that resets the data terminal ready condition (0-15; 1 is recommended).	DTRDLY	1
Number of idle time units (53.3 milliseconds each) needed to satisfy idle state time considerations (0-255; 38 is recommended minimum).	IDLETIME	38
Number of base time units (500 milliseconds each) to receive intelligible data (0-255).	NONPRDRCV	2
Number of retries to be performed before the line is considered inoperative (0-21).	RETRY	1
The line is to be varied online when CPF is started (*NO or *YES).	ONLINE	*YES
Valid only for nonswitched lines. List *on this work sheet only* (not on the CRTLIND command prompt) the name(s) of the control units to be attached to this line (up to 50). The normal order of configuring communications is CRTLIND, CRTCUD, then CRTDEVD. If you follow this order, when you create control units that reference this line (through the LINE parameter), the name of the control units are automatically inserted in the CTLU parameter for this line. (Use additional sheets if necessary.)	CTLU	RCU
The authority for this line description to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	*NORMAL
Brief description of the line description (*BLANK or no more than 50 characters in apostrophes.)	TEXT	

'Switched line to branch office'

5251 CONTROL UNIT
(CRTCUD command)

Description		Parameter	Entry
Name of the control unit.	R	CUD	RCU
Control unit type identifier (5251).	R	TYPE	5251
Model number of the control unit (2 or 12).	R	MODEL	12
Control unit address (see the appropriate Remote Work Station Configuration Work Sheet):	R	CTLADR	Ø1ØØ

Type of Line	Entry	
Switched	xx00, where xx =	01-FE and must be unique on your system. (For IBM 2400 or 4800 bps Integrated Modems, xx can be one of the following values: 04, 05, 06, 07, 08, 09, or xA, xB, xC, xD, xE, or xF, where x = 1-9.)·
Nonswitched	xxyy, where xx =	01-FE and must be unique on the line. (For IBM 2400 or 4800 bps Integrated Modems, xx can be one of the following values: 04, 05, 06, 07, 08, 09, or xA, xB, xC, xD, xE, or xF, where x = 1-9.)
	and yy =	LINNBR parameter value from CRTLIND work sheet.

Attached to a switched line (*NO or *YES).	SWITCHED	*YES
Name of the nonswitched line to which this control unit is attached (*NONE if attached to a switched line).	LINE	*NONE
The modem has the data rate select feature (*NO or *YES).	SELECT	*YES
Telephone number (4 to 16 digits) of this control unit. (See appropriate Remote Work Station Configuration Work Sheet.) Valid only for SWITCHED(*YES) or SWNBKU(*YES).	TELNBR	4Ø42389999
Method to be used to make the initial connection between a switched line and the control unit (*ANS or *CALL). Valid only for SWITCHED(*YES) or SWNBKU(*YES).	INLCNN	*CALL
Exchange identifier used to identify this control unit to the remote system or device (020000xx, where xx is the switch setting of the controller station address on the 5251). Valid only for SWITCHED(*YES) or SWNBKU(*YES).	EXCHID	Ø2ØØØØØ1
This control unit is to be varied online when CPF is started (*NO or *YES).	ONLINE	*YES
List of line names that identify the lines that can be connected to this control unit. Valid only for SWITCHED(*YES) or SWNBKU(*YES). **Note:** For each line name specified, a line description by that name must already exist.	LINLST	LINE2Ø
The modem has the switched network (dial) backup feature (*NO or *YES).	SWNBKU	*NO
If the connection with this control unit is delayed (for instance, if the 5251 Model 2 or 12 is powered off), the system attempts to make a connection periodically (*NO or *YES). Valid only for SWITCHED(*NO).	DLYFEAT	*NO
List *on this work sheet only* (not on the CRTCUD command prompt itself) the name(s) of the devices to be attached to this control unit (1-9 remote work stations; see *5250 Communications Network Setup Form*). *Do not enter values for the DEV parameter on the CRTCUD command prompt.* When you create individual device descriptions for communications devices, and you reference this control unit through the CTLU parameter, those device names are automatically inserted in the DEV parameter for this control unit.	DEV	RWSØ1 RWSPR1
(Use additional sheets if necessary.)		
The authority for this control unit to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	*NORMAL
Brief description of the control unit (*BLANK or no more than 50 characters in apostrophes.)	TEXT	

'Branch Office 5251-12 Control Unit'

DISPLAY STATION (PART 1 OF 2)
(CRTDEVD command)

Description		Parameter	Entry
Name of the display station. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)	R	DEVD	RWSØ1
Physical address of the device:	R	DEVADR	ØØØØØØ

Control Unit	Entry
WSC or WSCE	000000
5251	xxyyyy

yyyy — CTLADR parameter values from CRTCUD work sheet

xx — Unit address (00 if device is part of 5251 Model 2 or 12; 02-05 if attached to first cluster; 06-09 if attached to second cluster)

Description		Parameter	Entry
Device type (5251, 5252, 5291, or 5292).	R	DEVTYPE	5251
Device model:	R	MODEL	11

Device Type	Screen Size	Entry
5251	960 characters	1
	1920 characters	11
5252	960 characters (dual)	1
5291	1920 characters	1
5292	1920 characters	1

Description		Parameter	Entry
Name of associated work station controller or 5251 control unit. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)		CTLU	RCU
This device is varied online when CPF is started (*NO or *YES).		ONLINE	*YES

Work Sheet for a Display Station

DISPLAY STATION (PART 2 OF 2)
(CRTDEVD command)

Description	Parameter	Entry
The line connection (switched lines only) is to be broken after the work station user signs off without specifying a value for the DROP parameter on the SIGNOFF command (*NO or *YES).	DROP	*YES
Name of the associated work station printer (*NONE or device name).	PRINTER	RWSPR1
Name of an alternative printer file to be used when no associated work station printer is available.	PRTFILE	________
Address of device:	WSCADR	________

Control Unit	Entry
5251	000000
WSC or WSCE	xxyyzz

zz — Work station address switch settings (00-06)

yy — Work station controller port number as follows:

WSC	Valid Entries	WSCE	Valid Entries
WSC1	00-15	WSCE1	00-07
WSC2	16-31	WSCE2	16-23
WSC3	32-47	WSCE3	32-39
WSC4	48-63	WSCE4	48-55

xx — Unit address (00-19 if WSC; 00-31 if WSCE)

(See the appropriate *Local Work Station Configuration Work Sheet.*)

Description	Parameter	Entry
Type of keyboard (only when display station is connected to WSC or WSCE):	WSCKBD	________

Entry

yzzz

zzz — 3-character identifier (see CRTDEVD command in *CL Reference Manual*)

y — T for typewriter-like keyboard
D for data entry keyboard without proof arrangement
P for data entry keyboard with proof arrangement

Description	Parameter	Entry
Application program is to control blinking cursor (*YES or *NO).	ALWBLN	________
The authority for this device to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	*NORMAL
Brief description of the device. (*BLANK or no more than 50 characters, enclosed in apostrophes.)	TEXT	

'Branch Office Display Station'

____

Work Sheet for a Display Station

WORK STATION PRINTER (PART 1 OF 2)
(CRTDEVD command)

Description		Parameter	Entry
Name of the work station printer. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)	R	DEVD	RWSPR1
Physical address of the device:	R	DEVADR	Ø2Ø1ØØ

Control Unit	Entry
WSC or WSCE	000000
5251	xxyyyy

└── CTLADR parameter value from CRTCUD work sheet

└── Unit address (02-05 if attached to first cluster; 06-09 if attached to second cluster)

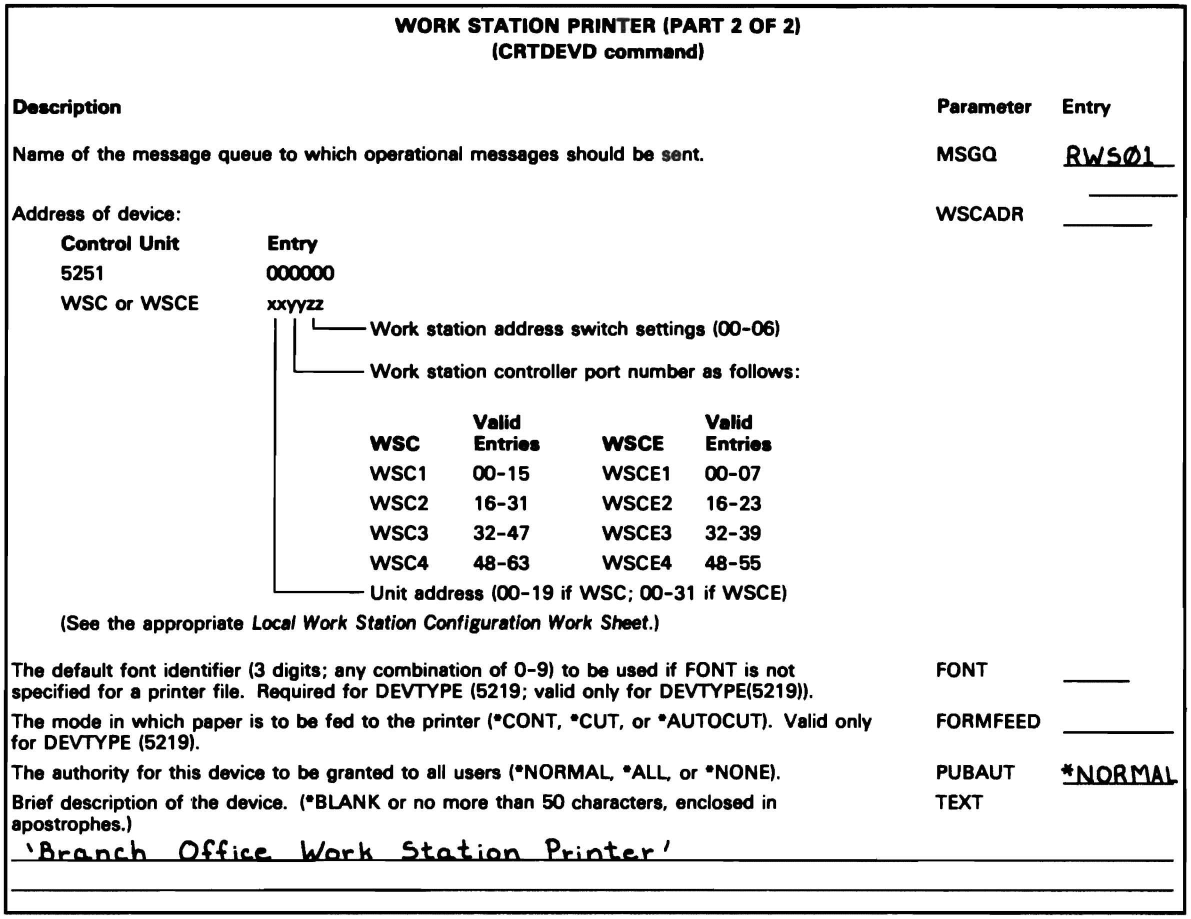
(See the appropriate *5250 Communications Network Setup Form.*)

Device type (5219, 5224, 5225, or 5256).	R	DEVTYPE	5256
Device model:	R	MODEL	2

Device Type	Model	Entry
5219	D1 D2	D1 D2
5224	1	1
	2	2
5225	1	1
	2	2
	3	3
	4	4
5256	1 2 3	1 2 3

Name of the associated work station controller or 5251 control unit. (See the appropriate *Local Work Station Configuration Work Sheet* or *5250 Communications Network Setup Form.*)		CTLU	RCU
The device is to be varied online when CPF is started (*NO or *YES).		ONLINE	*YES

Work Sheet for a Work Station Printer



WORK STATION PRINTER (PART 2 OF 2)
(CRTDEVD command)

Description	Parameter	Entry
Name of the message queue to which operational messages should be sent.	MSGQ	RWSØ1
Address of device:	WSCADR	

Control Unit	Entry
5251	000000
WSC or WSCE	xxyyzz

- zz — Work station address switch settings (00-06)
- yy — Work station controller port number as follows:

WSC	Valid Entries	WSCE	Valid Entries
WSC1	00-15	WSCE1	00-07
WSC2	16-31	WSCE2	16-23
WSC3	32-47	WSCE3	32-39
WSC4	48-63	WSCE4	48-55

- xx — Unit address (00-19 if WSC; 00-31 if WSCE)

(See the appropriate *Local Work Station Configuration Work Sheet.*)

Description	Parameter	Entry
The default font identifier (3 digits; any combination of 0-9) to be used if FONT is not specified for a printer file. Required for DEVTYPE (5219; valid only for DEVTYPE(5219)).	FONT	
The mode in which paper is to be fed to the printer (*CONT, *CUT, or *AUTOCUT). Valid only for DEVTYPE (5219).	FORMFEED	
The authority for this device to be granted to all users (*NORMAL, *ALL, or *NONE).	PUBAUT	*NORMAL
Brief description of the device. (*BLANK or no more than 50 characters, enclosed in apostrophes.)	TEXT	

'Branch Office Work Station Printer'

Work Sheet for a Work Station Printer

Library List Considerations

When setting up a library list:

- The libraries must exist on the system.
- A library name in QSYSLIBL/QUSRLIBL is ignored at CPF start-up, if the library does not exist.
- Library names in QSYSLIBL/QUSRLIBL cannot be deleted after CPF is started.
- Non-existing library prevents the job from starting.
- User must have “operational rights” to the library or the job is not started.

Operational rights to a library are to:

- Use a library.
- Display the description of a library.
- Place a library in the library list.

One of the ways to implement security on the System/38 is to deny users operational rights to specific libraries.

Progress Check – Unit 2: Installing Your System

Read each question carefully. Record your answers in the space provided.

1. How do you make a permanent change to the library list? When does the change take effect?

   A The Change System Value (CHGSYSVAL) command makes a permanent change to the library list.

   B. The change takes effect on new jobs that are started after the system values are changed.

2. Can you create a device description for a control unit that is active?

   No. You must take (vary) the control unit offline.

3. In what sequence should the descriptions be created for a remote configuration?

   A line descriptions
   B control unit descriptions
   c device description

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 3: Post-installation

Read each question carefully. Record your answers in the space provided.

1. What is the purpose of the Add Work Station Entry (ADDWSE) command?

The Add Work Station Entry (ADDWSE)
command adds a work station job
entry to the specified subsystem
description. The purpose of this command
is to identify the work station (s)
by name or type to the subsystem.
The work stations identified are
allowed to sign on to the subsystem
and perform work.

2. How is the system saved and how can the system be restored?

The system is saved with Save System
(SAVSYS) command. Other objects are saved
with Save Library (SAVLIB) command
specifying LIB(*NONSYS).
The system is restored using the CPF
installation procedures(steps).

3. Why should the security officer's password be changed?

The security officer's profile allows
anyone to perform most operations
on objects in the system. For that reason,
the profile should be limited to one person'

4. How can you prevent service dumps and job logs from being printed with other output reports?

Create output queues for service dumps
and job logs. Change the print files
(QPSRVDMP and QPJOBLOG) to use the
new queues

5. Can you add a work station to an active subsystem?

No. You terminate the subsystem.
The Terminate Subsystem (TRMSBS) command
terminates a subsystem or all subsystems.

6. What must be done when a work station is changed to another port on the work station controller?

Delete the current device description.
Create a new device description with
the new address. You cannot use the
Change Device Description (CHGDEVD)
to change the work station address.

7. What format must diskettes have in order to save the system?

Diskettes must be in the save/restore
(*SAVRST) format in order to save
the system or any object.

When you are finished, return to the module and review your answers with those in the text.

Module 1 Machine Exercise – Displaying Device Configuration

The purpose of this machine exercise is to allow you to use Control Language commands to:

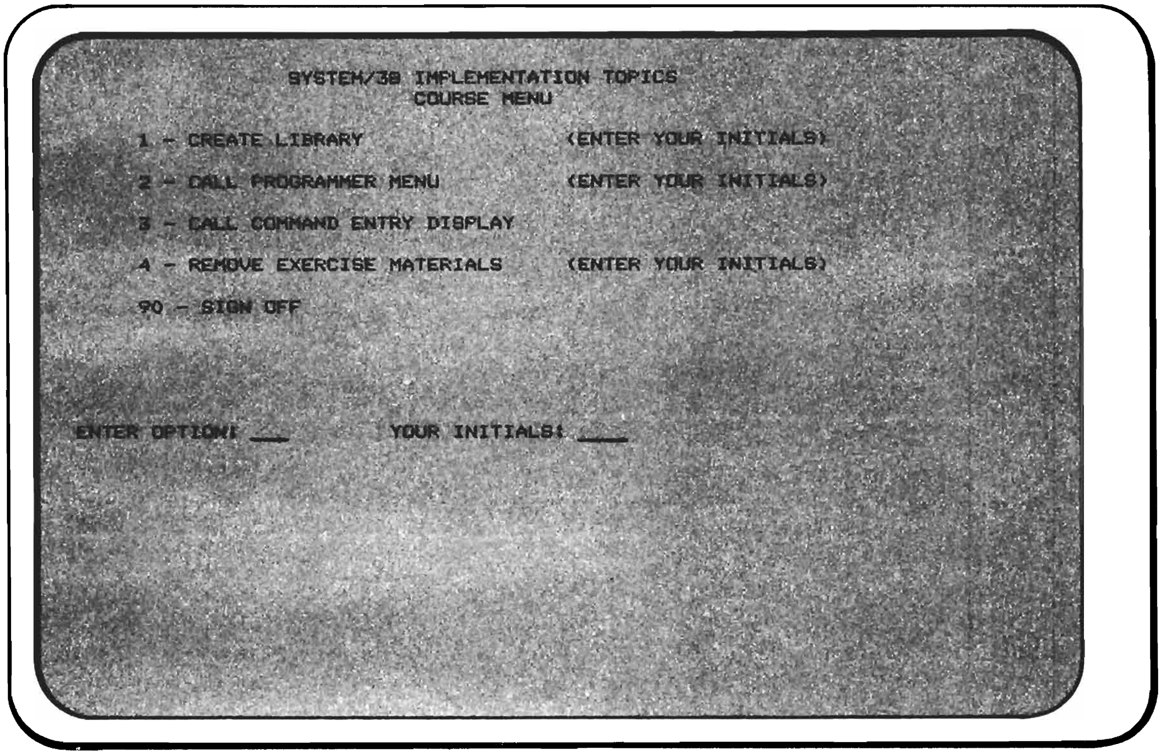
- Display the device configuration
- Display the control unit status for QWSC1
- Display the device description for your work station

The exercise requires that you display and/or list the system devices, local work stations, and remote devices (if any) to provide historical information. Also, these commands can be used to determine the current configuration of your system.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

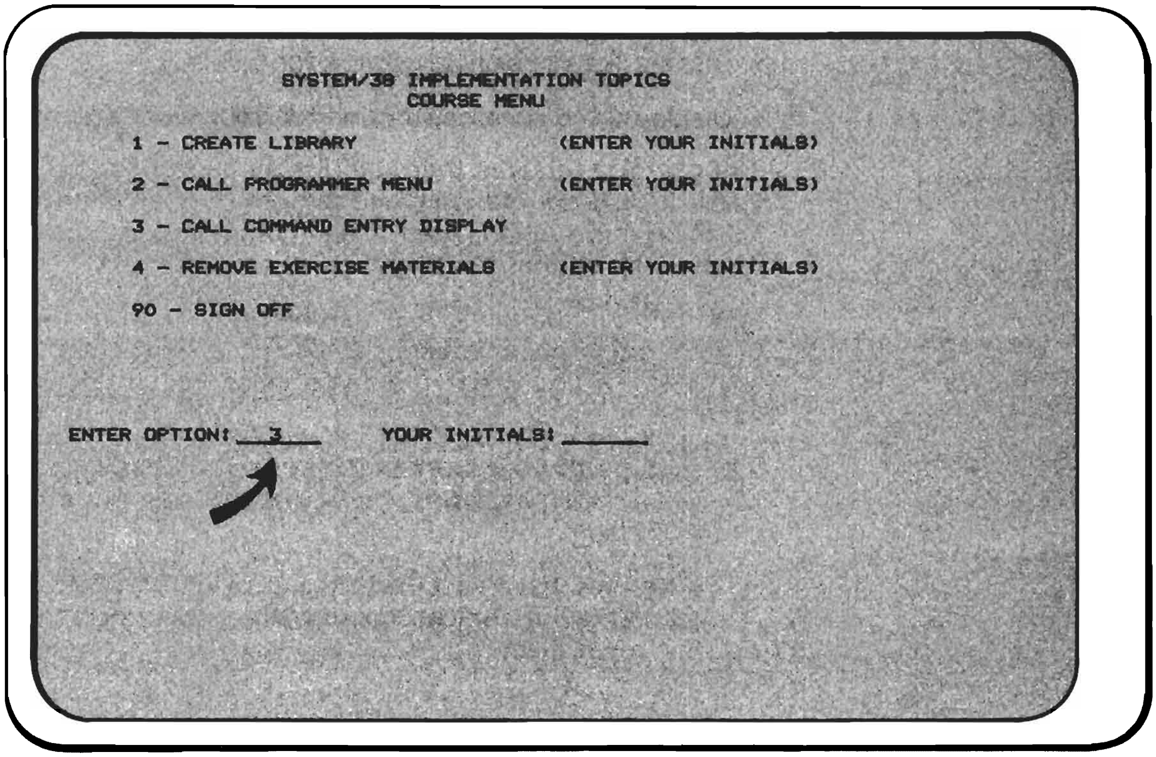
1. Sign on to your work station using the password S38IMPL. This is the password you will use to sign on to begin each exercise.

   After you have successfully signed on, the System/38 Implementation Topics Course Menu appears on the screen. It is shown below.



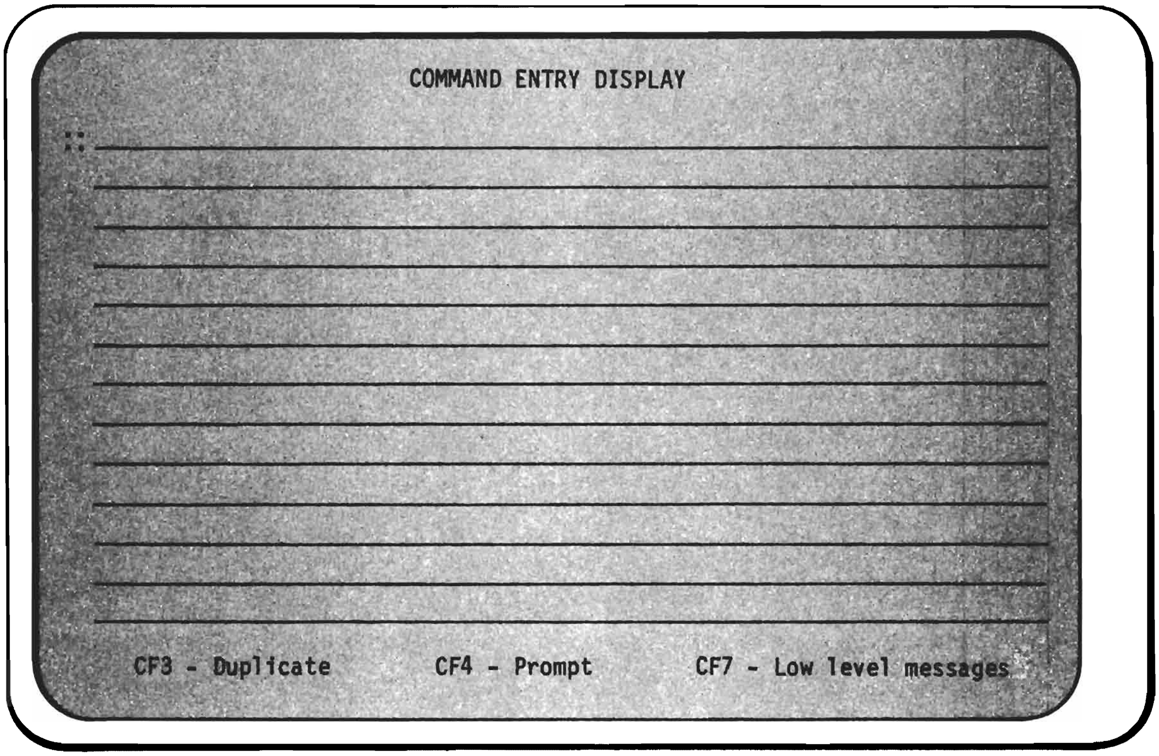
```
                SYSTEM/38 IMPLEMENTATION TOPICS
                          COURSE MENU

    1 - CREATE LIBRARY                  (ENTER YOUR INITIALS)

    2 - CALL PROGRAMMER MENU            (ENTER YOUR INITIALS)

    3 - CALL COMMAND ENTRY DISPLAY

    4 - REMOVE EXERCISE MATERIALS       (ENTER YOUR INITIALS)

    90 - SIGN OFF



ENTER OPTION: ___         YOUR INITIALS: ___
```

2. On the System/38 Implementation Topics Course Menu, select option 3 to obtain the Command Entry Display. (Note: To select option 3, key a 3 and press the Enter key.)



```
                  SYSTEM/38 IMPLEMENTATION TOPICS
                            COURSE MENU

     1 - CREATE LIBRARY                    (ENTER YOUR INITIALS)

     2 - CALL PROGRAMMER MENU              (ENTER YOUR INITIALS)

     3 - CALL COMMAND ENTRY DISPLAY

     4 - REMOVE EXERCISE MATERIALS         (ENTER YOUR INITIALS)

     90 - SIGN OFF




 ENTER OPTION: __3__     YOUR INITIALS: ______
```

You will use the Command Entry Display to enter commands for this exercise. It is shown below.

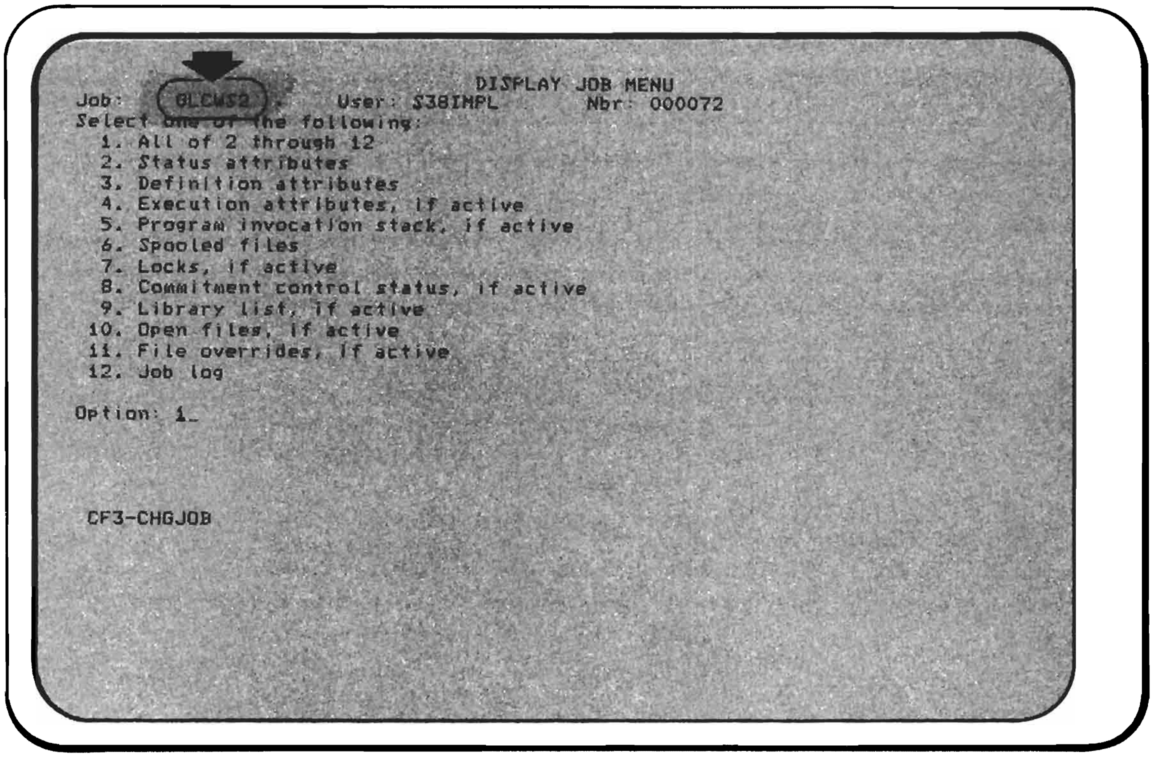


COMMAND ENTRY DISPLAY

CF3 - Duplicate CF4 - Prompt CF7 - Low level messages

3. Determine the name of your work station by entering the CL command DSPJOB and pressing the Enter key.

   A Display Job Menu, similar to the one below, appears. The work station name is the same as the job name. Its location is shown on the example below. Note the name for **your** work station shown on your display.

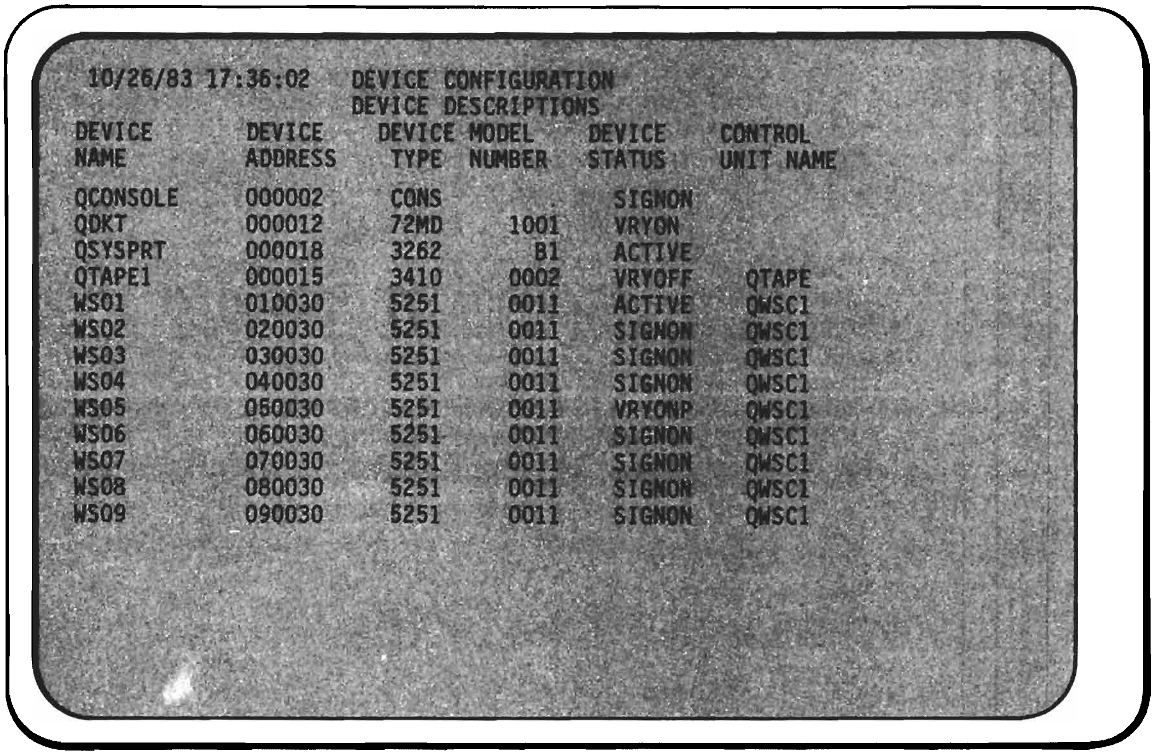


```
                              DISPLAY JOB MENU
Job:    BLCWS2 .      User: S38IMPL       Nbr: 000072
Select one of the following:
  1. All of 2 through 12
  2. Status attributes
  3. Definition attributes
  4. Execution attributes, if active
  5. Program invocation stack, if active
  6. Spooled files
  7. Locks, if active
  8. Commitment control status, if active
  9. Library list, if active
 10. Open files, if active
 11. File overrides, if active
 12. Job log

Option: 1_



 CF3-CHGJOB
```

4. After you have noted the name of your work station, press CF 1 (Command Key 1) to return to the Command Entry Display.

5. On the Command Entry Display, enter the Display Device Configuration command, DSPDEVCFG, to display the device configuration of the Guided Learning Center's System/38. For the details on the format of this command, see the module text or the System/38 Control Language Reference Manual.

6. The initial display is for the control unit descriptions. Press the Enter key to continue. The Device Configuration Device Descriptions display appears on the screen. It should be similar to the one shown below. (Note: If the System/38 you are using has remote devices, you must press the Enter key one or more times until the Device Configuration Device Descriptions display appears.)



```
  10/26/83 17:36:02   DEVICE CONFIGURATION
                      DEVICE DESCRIPTIONS
DEVICE       DEVICE    DEVICE MODEL    DEVICE     CONTROL
NAME         ADDRESS    TYPE  NUMBER   STATUS     UNIT NAME

QCONSOLE     000002     CONS             SIGNON
QDKT         000012     72MD    1001     VRYON
QSYSPRT      000018     3262      B1     ACTIVE
QTAPE1       000015     3410    0002     VRYOFF     QTAPE
WS01         010030     5251    0011     ACTIVE     QWSC1
WS02         020030     5251    0011     SIGNON     QWSC1
WS03         030030     5251    0011     SIGNON     QWSC1
WS04         040030     5251    0011     SIGNON     QWSC1
WS05         050030     5251    0011     VRYONP     QWSC1
WS06         060030     5251    0011     SIGNON     QWSC1
WS07         070030     5251    0011     SIGNON     QWSC1
WS08         080030     5251    0011     SIGNON     QWSC1
WS09         090030     5251    0011     SIGNON     QWSC1
```

Look for your work station under the heading "Device Name" on the display. Under the heading "Control Unit Name", determine the controller to which your work station is attached.

Write down the name of the Control Unit for your work station. ______________

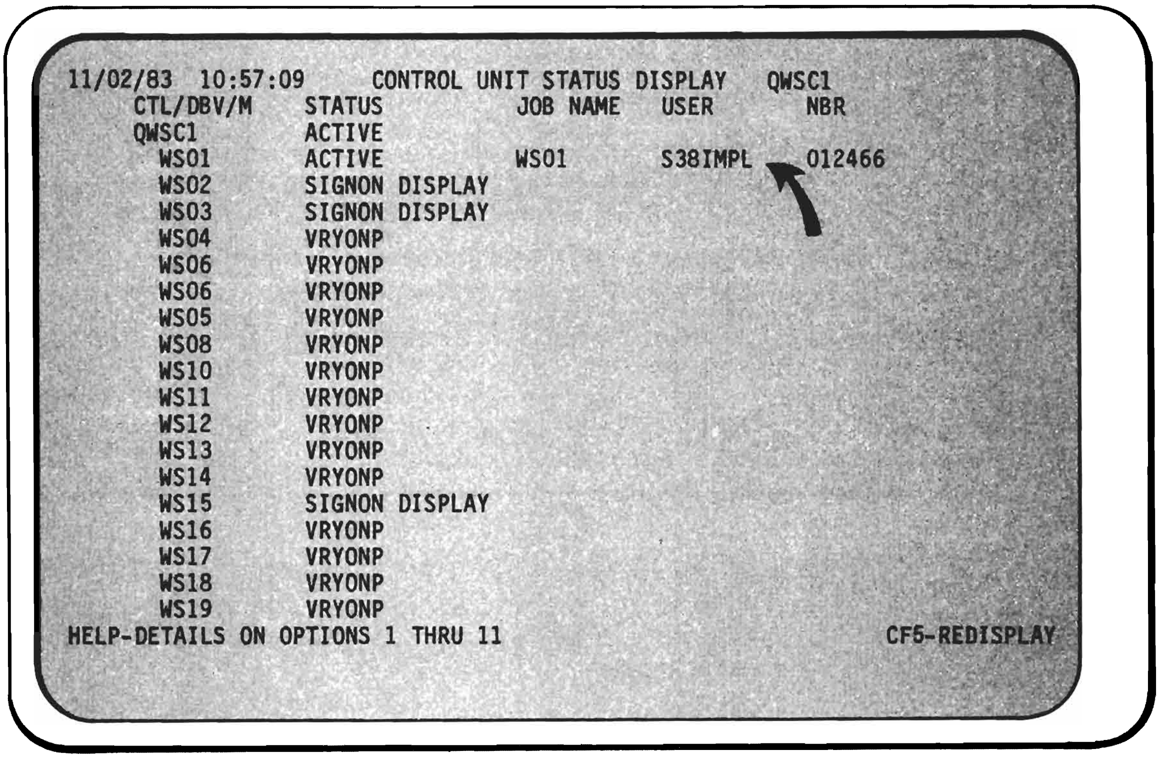
The "Device Status" column indicates your work station is active – it is online and in use. Other status codes you may see are:

- VRYOFF – The device has been varied offline
- VRYON – The device is varied online
- VRYONP – The device is being varied online
- SIGNON – The device has a sign on display

7. When you have recorded the name of the Control Unit for your work station, press the Enter key to return to the Command Entry Display.

8. Display the configuration (control unit status) of the control unit for your work station with the DSPCTLSTS command. See the module text or the System/38 Control Language Reference Manual for the format of this command. (Note: If you are using a remote work station, do not specify a control unit name with the command.)

   Your display should be similar to the one below.

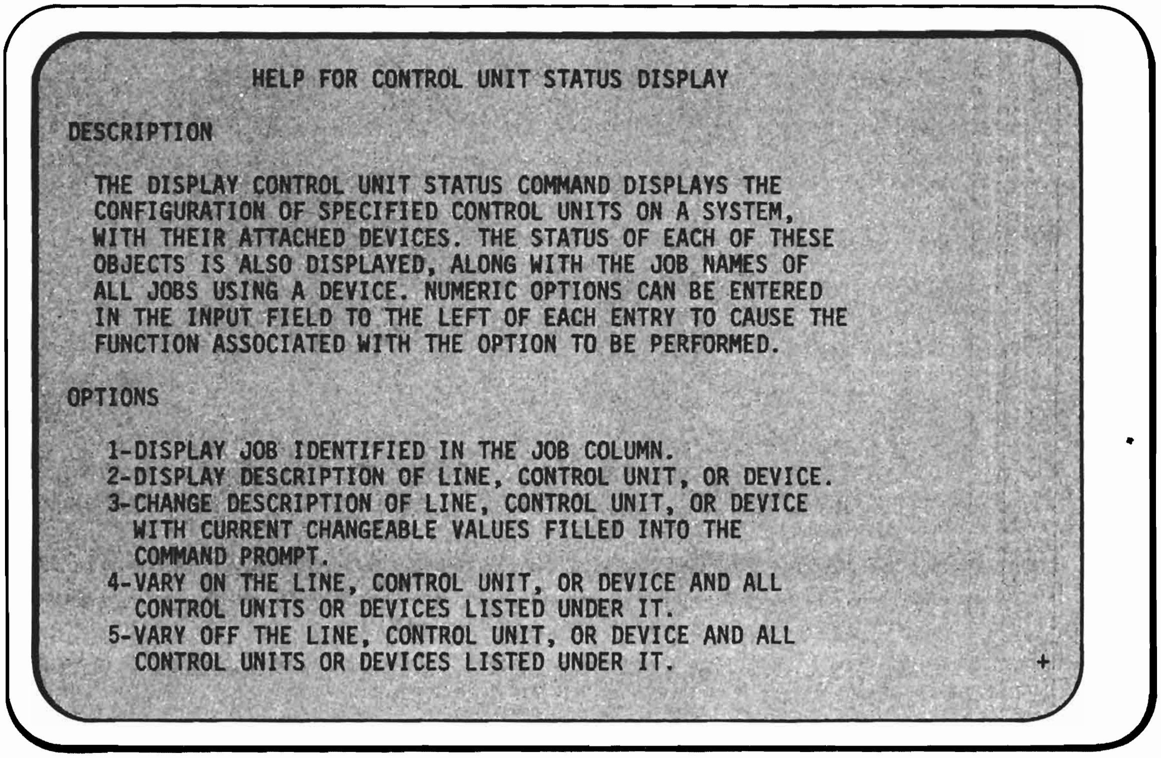


```
11/02/83  10:57:09      CONTROL UNIT STATUS DISPLAY   QWSC1
     CTL/DBV/M     STATUS              JOB NAME   USER       NBR
     QWSC1         ACTIVE
       WS01        ACTIVE              WS01       S38IMPL    012466
       WS02        SIGNON DISPLAY
       WS03        SIGNON DISPLAY
       WS04        VRYONP
       WS06        VRYONP
       WS06        VRYONP
       WS05        VRYONP
       WS08        VRYONP
       WS10        VRYONP
       WS11        VRYONP
       WS12        VRYONP
       WS13        VRYONP
       WS14        VRYONP
       WS15        SIGNON DISPLAY
       WS16        VRYONP
       WS17        VRYONP
       WS18        VRYONP
       WS19        VRYONP
HELP-DETAILS ON OPTIONS 1 THRU 11                              CF5-REDISPLAY
```

Locate your device by work station name. The following information is associated with your work station:

- Status – ACTIVE
- Job Name – the same as your work station name
- User – S38IMPL

9. Press the HELP key to obtain the "Help For Control Unit Status Display." Your display should be similar to the one below.



```
              HELP FOR CONTROL UNIT STATUS DISPLAY

DESCRIPTION

  THE DISPLAY CONTROL UNIT STATUS COMMAND DISPLAYS THE
  CONFIGURATION OF SPECIFIED CONTROL UNITS ON A SYSTEM,
  WITH THEIR ATTACHED DEVICES. THE STATUS OF EACH OF THESE
  OBJECTS IS ALSO DISPLAYED, ALONG WITH THE JOB NAMES OF
  ALL JOBS USING A DEVICE. NUMERIC OPTIONS CAN BE ENTERED
  IN THE INPUT FIELD TO THE LEFT OF EACH ENTRY TO CAUSE THE
  FUNCTION ASSOCIATED WITH THE OPTION TO BE PERFORMED.

OPTIONS

   1-DISPLAY JOB IDENTIFIED IN THE JOB COLUMN.
   2-DISPLAY DESCRIPTION OF LINE, CONTROL UNIT, OR DEVICE.
   3-CHANGE DESCRIPTION OF LINE, CONTROL UNIT, OR DEVICE
     WITH CURRENT CHANGEABLE VALUES FILLED INTO THE
     COMMAND PROMPT.
   4-VARY ON THE LINE, CONTROL UNIT, OR DEVICE AND ALL
     CONTROL UNITS OR DEVICES LISTED UNDER IT.
   5-VARY OFF THE LINE, CONTROL UNIT, OR DEVICE AND ALL
     CONTROL UNITS OR DEVICES LISTED UNDER IT.                +
```

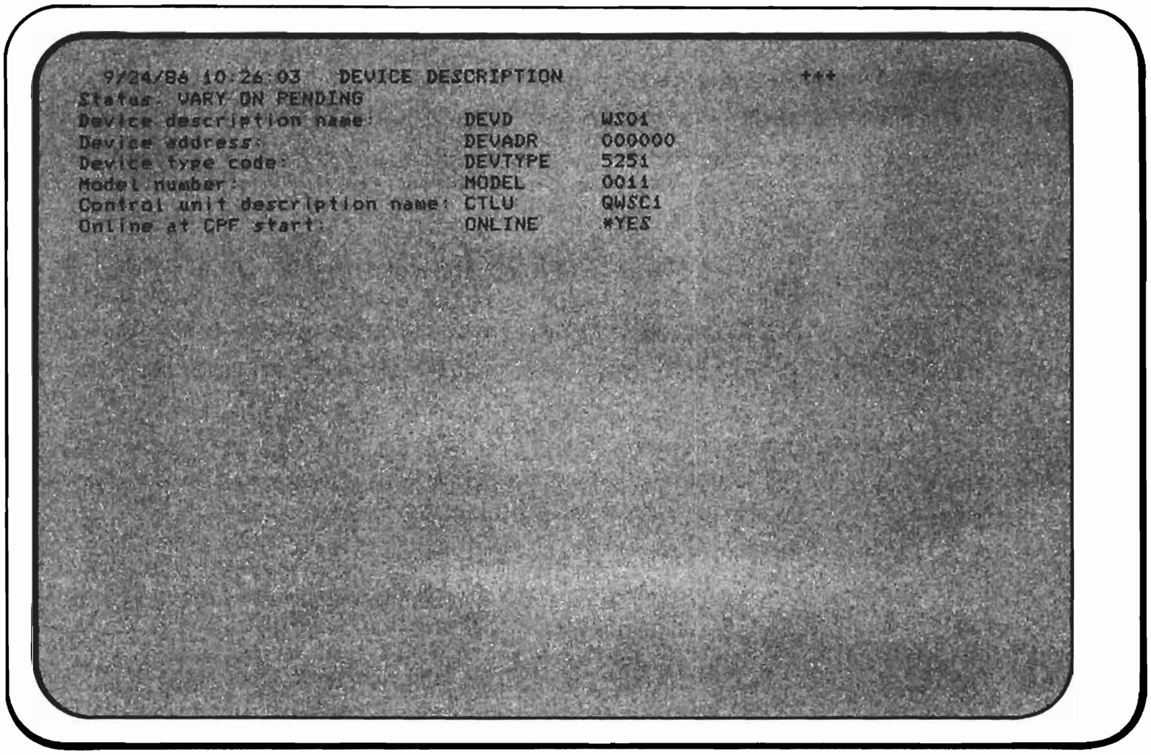
10. From the "Help For Control Unit Status Display", determine the option to display a device description.

    Write down the option number to display a device description. ______

11. Press the Enter key to return to the "Control Unit Status Display."

12. Display the device description for your work station. To do so, position the cursor on the line with your work station's name. Type the option number you identified in Step 10. Press the Enter key.

    Your display should be similar to the one shown below.

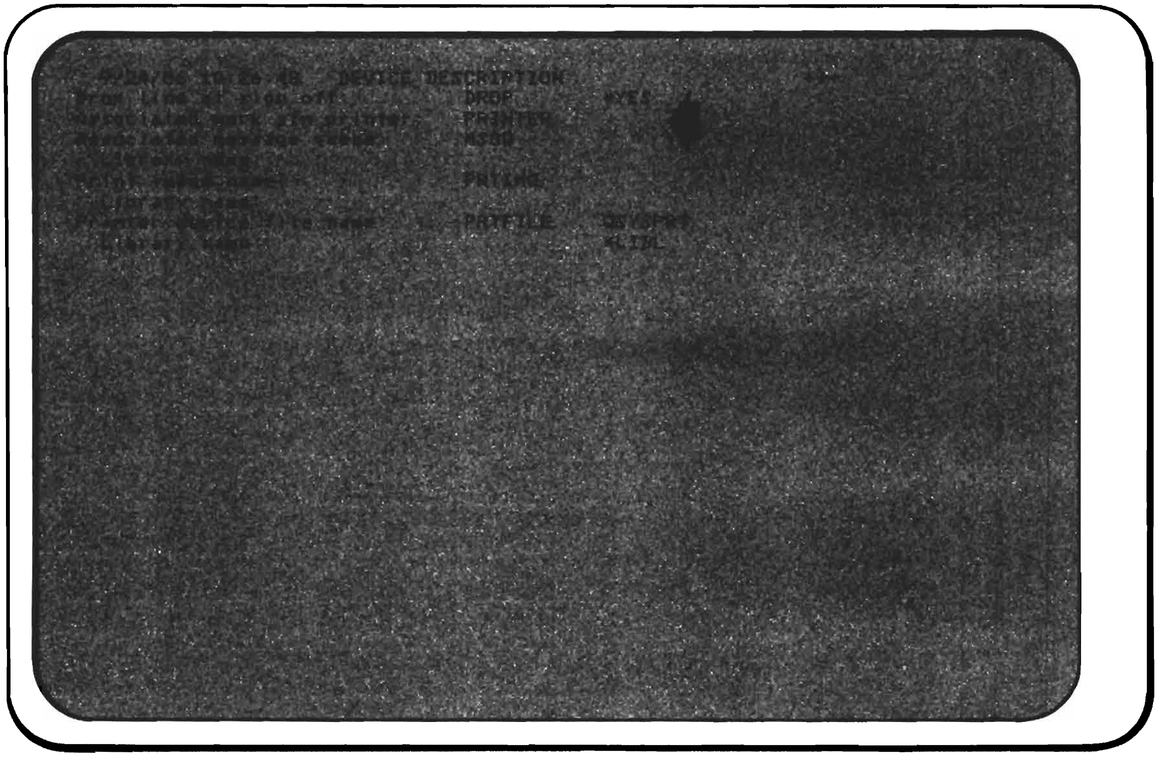


```
  9/24/86 10:26:03   DEVICE DESCRIPTION                        +++
Status: VARY ON PENDING
Device description name:        DEVD        WS01
Device address:                 DEVADR      000000
Device type code:               DEVTYPE     5251
Model number:                   MODEL       0011
Control unit description name: CTLU        QWSC1
Online at CPF start:            ONLINE      *YES
```

Note that with this display you can actually verify what has been configured for the device. You can check such items as:

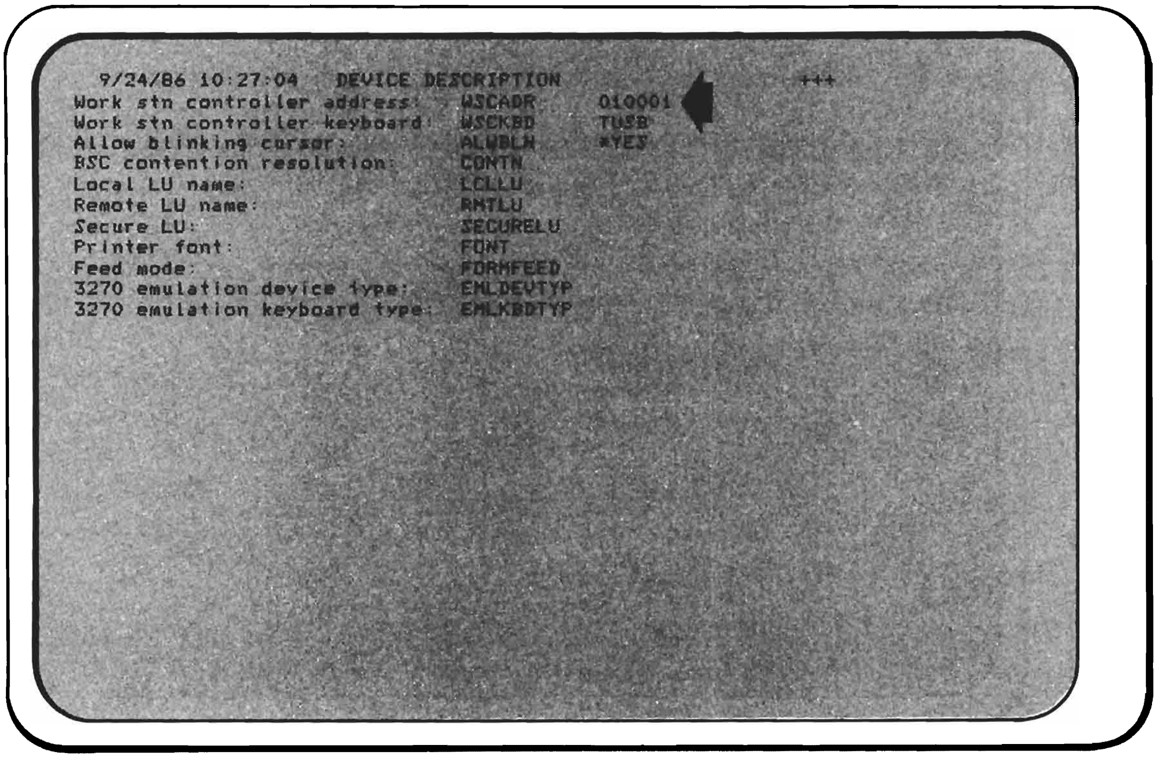
- Name
- Address
- Device type
- Device model
- Control unit
- Whether the device is to be placed online at CPF startup.

13. Press the Enter key once to obtain the next page of the "Device Description" display. This display should be similar to the one below and show, among other things, whether your work station has an associated work station printer.



```
 [illegible]        DEVICE DESCRIPTION
Drop line at sign off              DROP      *YES
[illegible]                        PRINTER
[illegible]                        MSGQ
  [illegible]
[illegible]                        PRTIMG
  [illegible]
Printer device file name           PRTFILE   QSYSPRT
  Library name                               *LIBL
```

14. Press the Enter key one more time to get the next page of the ''Device Description'' display. Note the entry ''Work Stn Controller Address.'' It shows the device address, work station controller port number, and the settings of the address switches on the device.

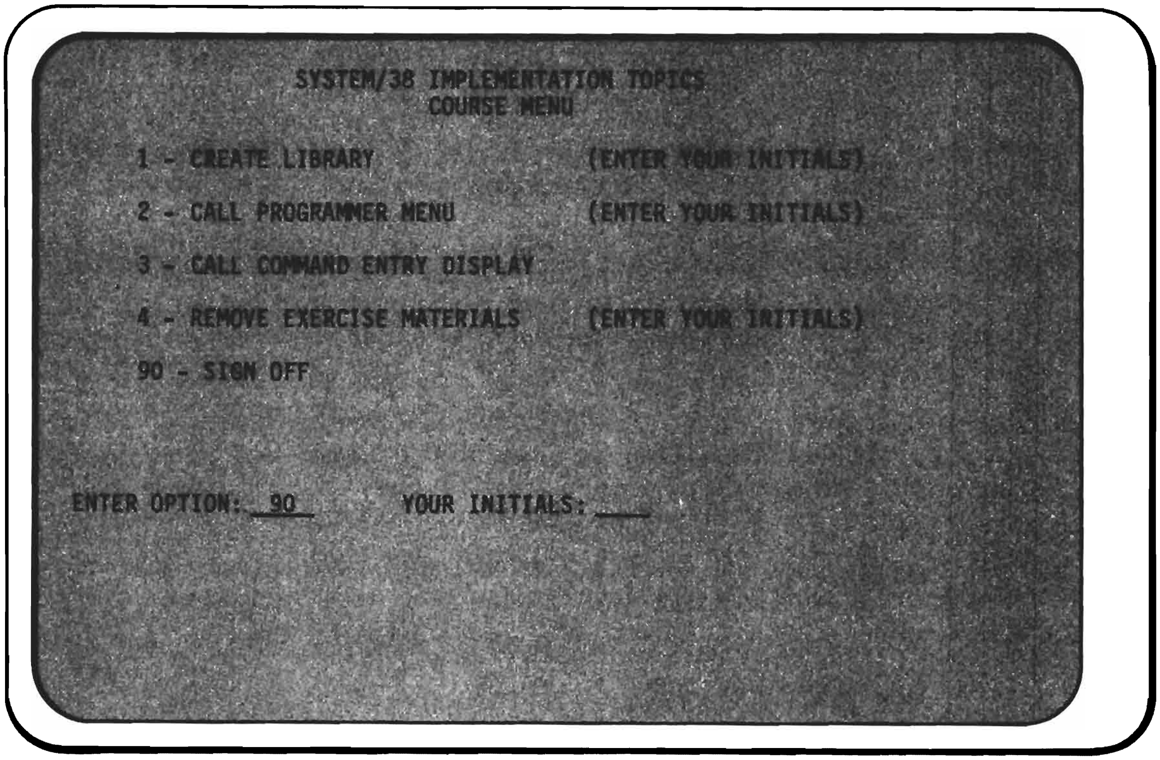


```
  9/24/86 10:27:04   DEVICE DESCRIPTION                  +++
Work stn controller address:    WSCADR      010001
Work stn controller keyboard:   WSCKBD      TUSB
Allow blinking cursor:          ALWBLN      *YES
BSC contention resolution:      CONTN
Local LU name:                  LCLLU
Remote LU name:                 RMTLU
Secure LU:                      SECURELU
Printer font:                   FONT
Feed mode:                      FORMFEED
3270 emulation device type:     EMLDEVTYP
3270 emulation keyboard type:   EMLKBDTYP
```

15. Press CF 1 to return to the Command Entry Display.

16. Return to the System/38 Implementation Topics Course Menu by entering the command RETURN or pressing CF 1.

17. From the System/38 Implementation Topics Course Menu, select option 90 to sign off the work station.



```
                SYSTEM/38 IMPLEMENTATION TOPICS
                         COURSE MENU

   1 - CREATE LIBRARY                (ENTER YOUR INITIALS)

   2 - CALL PROGRAMMER MENU          (ENTER YOUR INITIALS)

   3 - CALL COMMAND ENTRY DISPLAY

   4 - REMOVE EXERCISE MATERIALS     (ENTER YOUR INITIALS)

   90 - SIGN OFF



ENTER OPTION: _90_      YOUR INITIALS: ____
```

This completes the machine exercise for Module 1 – System Installation. Continue with the Exercise Summary on the next page.

Exercise Summary

You have used commands to verify existing names of the device descriptions (configuration objects) for the system configuration once CPF is installed.

You can use the Display Device Configuration (DSPDEVCFG) command or Display Control Unit Status (DSPCTLSTS) command to display the device names assigned to a particular control unit. You can also use the Display Device Status (DSPDEVSTS) to display the status of one or all devices on the system.

You can use the Display Device Description (DSPDEVD) command to display the parameter values specified for a particular device description, or enter a 2 (as you have done) for the device description you want to display when using the DSPCTLSTS or DSPDEVSTS command.

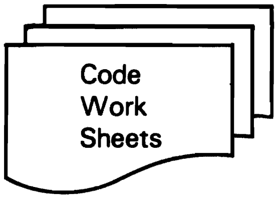
These commands can be used to completely fill in your work sheets, if you have not done so. Again, these sheets should be retained and kept updated for reference, as a guide for service personnel, in the event of expansion or relocation.

At this time, continue with the Module 1 Summary on the next page.

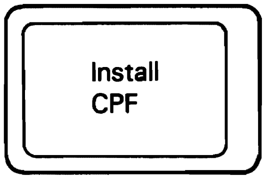
Module 1 Summary

Your Customer Service Representative (CSR) has the responsibility for setting up the system unit and all the systems devices. The CSR executes diagnostic programs to ensure the hardware is working properly.

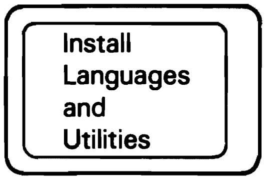
You have been taught the steps to install your System/38. They are:



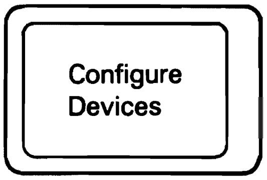
Plan for device configuration – code system configuration work sheets.



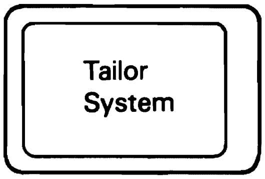
Install the Control Program Facility.



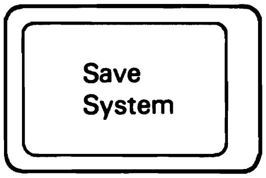
Install and verify the languages and utilities.



Describe the configuration with CL commands.



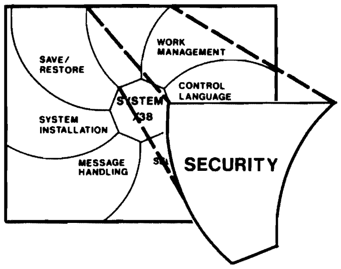
Modify the system to your specific needs.



Save your system.

Continue now with the Introduction to Module 2, Security, on the next page.

Module 2. Security



Now that you have installed your System/38, the next step is to secure the system.

The major requirement in the area of security is to prevent unauthorized access to data. Another requirement is to ensure data integrity. That is, you must be sure your data is updated only by authorized persons. You may have operators located in remote areas of the business who may view the work station as their own personal computer. Work stations can be used by persons who might want to retrieve or alter data which they should not be authorized to access. Module 2 shows you how to address these requirements of the System/38.

How much security do you need? If you have too much, you and the system become unproductive. If you have none or very little, any person can perform most functions after merely signing on. You need to decide what level of security provides the best results for your needs.

Perhaps the reason some companies don't protect their information more carefully is that they don't know what it is worth. Obviously, some companies (such as financial institutions) can place a value on account information and figure how much money is lost when such data are tampered with. But it is less clear as to what is the value of an inventory file.

A good guideline is to estimate how much business could be lost if those files are damaged, or how much it would cost to reconstruct the files. Using this rule, you would quickly realize that you can't be without your files without experiencing financial strain.

You will use the following materials for study of this module:

- Module 2 text
- IBM System/38 Programming Reference Summary (SC21-7734)
- IBM System/38 Control Program Facility Programmer's Guide (SC21-7730)

Begin your study of Security in the Module 2 text now.

STUDENT NOTES: Module 2. Security

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

1. CPF takes the password you type and checks it against the authorized user table. This table contains the list of passwords and the associated profiles for all system users. The table is updated by the system when you change passwords and when you add or delete a user profile.

The security officer can display the user profile and their associated passwords. This is password security for one level sign-on.

Two level sign-on uses both passwords and user ID. User ID's must be unique; however, passwords can be used for group profiles. One way encryption is used to store the passwords. With two level sign-on the work station operator must enter the correct password and user ID to gain access to the system. The security officer can display the date a password was changed but not the password.

The user profile is an object containing ① a description of the user 2. a list of objects owned by the user 3. and a list of user authorizations to objects and their functions.

The Create User Profile (CRTUSRPRF) command identifies an authorized user to the system and creates a user profile for that user.

User profiles are easily created and maintained through the use of certain CL commands. These commands are authorized only to the security officer.

STUDENT NOTES (continued): Security

1 is the highest priority

Private authorized objects are available only to specific users.

Public objects are available to all users and do not appear in the user profile.

If the password is not coded, the name of the user profile becomes the password.

You can authorize specific use of any object you own.

Progress Check – Unit 1: Physical Security

Read each question carefully. Record your answers in the space provided.

1. Describe the 3 measures of physical security.

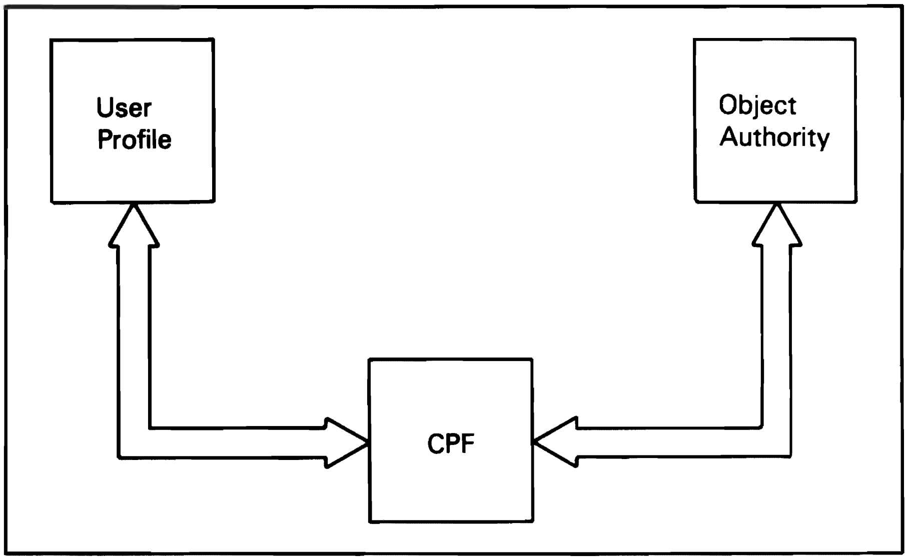
   1. Limit the access to your computer room.
   2. Limit the access to your system and work stations
   3. Limit the operation of your system and work stations. (VARY DeVICE COMMAND)

2. How can you place a device offline or online?

   You can use the Vary Device (VRYDEV) command. The VAYDEV command is used to vary one or more devices online with the system, or offline.

When you are finished, return to the module and review your answers with those in the text.

SECURITY COMPONENTS



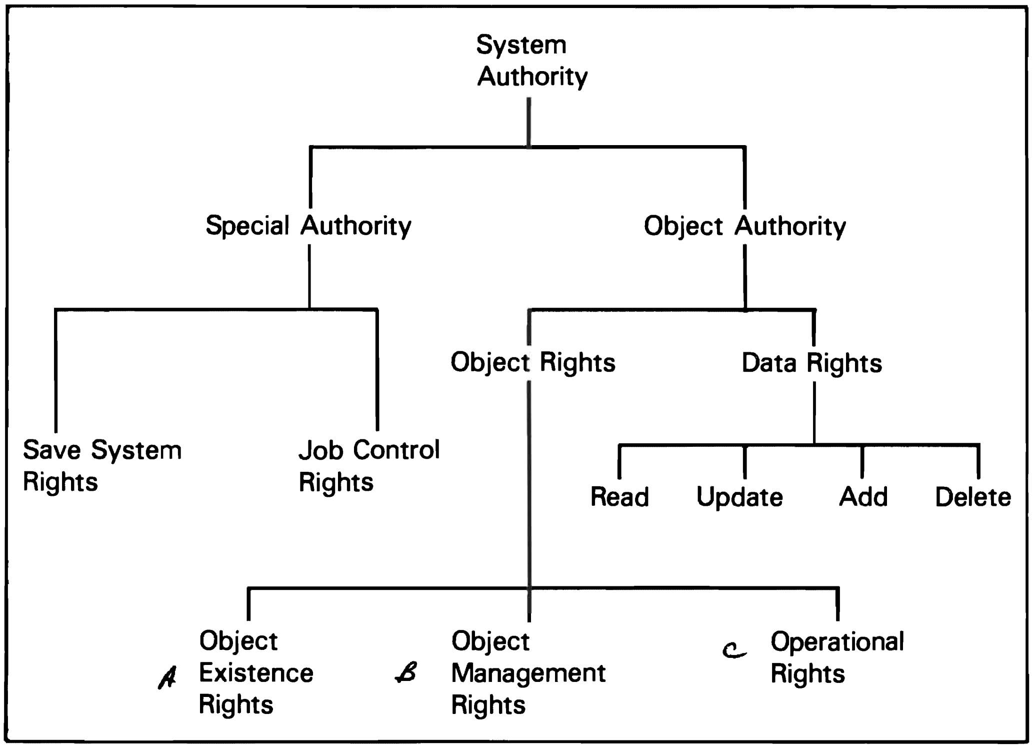
User profile - An object that contains the description of the user and a list of authorizations and functions.

Object Authority - The right to use or control an object.

There are two types of authority on the System/38 – special authority and object authority.

Special authority gives you save system rights and/or job control rights.

Object authority gives you the right to use or control an object.



A Existence - The right to delete, save, free the storage of, restore, and transfer the ownership of an object. For example, you must have object existence rights to a file to delete that file.

B. Management - The right to move, rename, grant authority to, revoke authority from, and change the attributes of an object.

C. Operational - The right to use an object and look at its description.

Progress Check – Unit 2: System Security

Read each question carefully. Record your answers in the space provided.

1. List the 2 major components of System/38 security.

   (1) user profile
   (2) object authority

2. What is special authority?

   Special authority is the right to perform certain system control operations. These operations are save system rights and job control rights. It does not, however, give the user the ability to access or use individual objects in any other way.

3. Describe object authority.

   Object authority is the right to use or control an object, such as reading a file or deleting a program. The owner of an object has complete authority over the object. There are two major groups of object authority rights - object rights + data rights.

4. List the 6 IBM-supplied user profiles and their passwords.

   1 QSECOFR --SECOFR (security officer)
   2 QPGMR - PGMR (programmer)
   3 QSYSOPR - SYSOPR (system operator)
   4 QUSER - USER (user)
   5 OPSR - PSR (programming service representative
   6 QCE - CE (customer engineer)

When you are finished, return to the module and review your answers with those in the text.

Module Summary

Unless keys, combinations, and passwords are changed with reasonable frequency, once they are compromised, they are compromised for significant periods of time. Further, reasonably frequent changes of such items reaffirms to all personnel the continuing concern for security.

In addition to requiring a password to sign on, you could have the initial program in the user profile require such personal information as a name or special identification code before using the system.

Limit the number of attempts that anyone can make in logging on to the system, in order to limit or deter any guessing.

Keep a record of all user activity in the system and monitor the history log to detect any unusual activity, such as repeated failures by someone to enter a correct password.

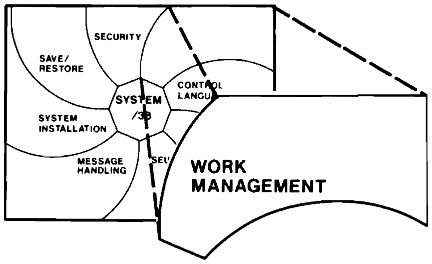
Keep in mind that your installation's security needs should be considered whenever application programs are designed. Some typical security considerations are

- System users should have access only to the functions and data needed to perform their job.
- Work station users should be able to access and update data in the data base only through tested procedures and programs.

You have seen the functions and procedures for implementing security on your system. You can implement the security function easily by using the system-supplied user profiles and defaults, but do not forget to change the passwords. As your applications require a higher level of security, you can use the more comprehensive security measures.

Continue now with the Introduction to Module 3, Work Management, on the next page.

Module 3. Work Management



Now that you know how to install and implement security procedures for your System/38, the next step is to see how work is performed on the system.

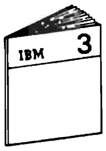
The ability to control the concurrent execution of different units of work in different environments is provided by the work management facilities.

Work is defined to the system as a job. Within a job, any number of related or unrelated functions can be performed. Thus a job is simply made up of whatever sequence of processing actions you want to perform. As a job is initiated for execution within the system, a set of parameters is defined to control the execution of that job.

All jobs processed in the system execute within an operating environment called a subsystem. You may think of a subsystem as a segment of main storage where your work is performed.

You will use the following materials for study of this module:

- Module 3 text
- IBM System/38 Control Program Facility Programmer's Guide (SC21-7730)
- IBM System/38 Control Language Reference Manual (SC21-7731)



Begin your study of Work Management in the Module 3 text now.

STUDENT NOTES: Module 3. Work Management

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

Work management controls the submission, initiation, execution, and termination of work.

5 Types of job

A System jobs
1. – spooling

B User Jobs
autostart
interactive
batch
communication

An interactive job starts when you sign on at a work station or the system console and ends when you sign off. Normally, an interactive job is one that requires operator interaction. Inquiry, data entry, and update are examples of interactive jobs.

A batch job starts when your job is selected from a job queue and ends when the program(s) terminates, if there is no spooled output. If there is spooled output, the job ends when the data is written from the output queue.

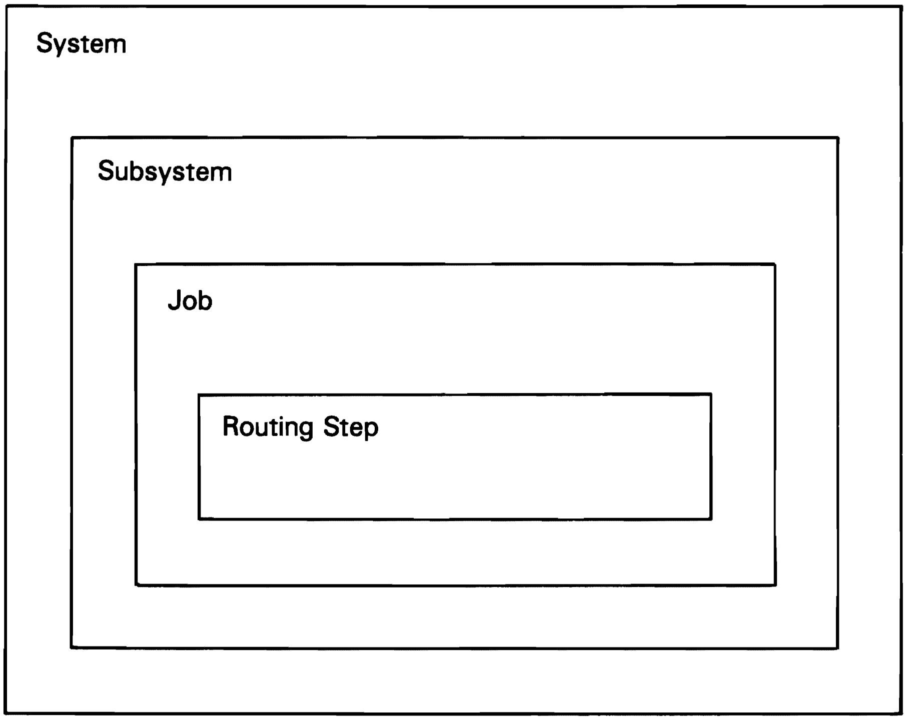
An autostart job is associated with a subsystem and starts whenever the subsystem is started and ends when its function terminates. A typical use of an autostart job is for one-time type work such as starting other subsystems or starting the system printer.

A communication job is one that is initiated from another System/38. This type of job is similar to a batch job except that no job queue is involved. Only commands valid in batch jobs can be executed in a communication job. A communication job is basically a batch job initiated by another system.

Work management objects provided with CPF

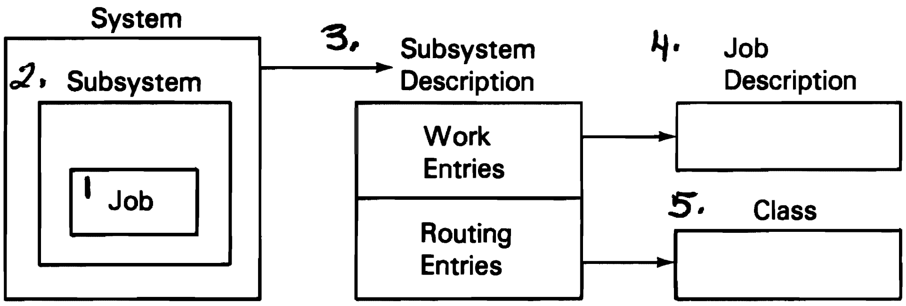
1. Job descriptions
2. subsystem descriptions
3. classes
4. Job queues
5. Output queues
6. profiles
7. programs.

SYSTEM/38 OPERATING ENVIRONMENT



- The System – contains system values that affect system operations. Examples of system values are date, time, and name of the controlling subsystem.
  - Subsystem – provides the predefined operating environment through which CPF processes work.
    - Job – work being or to be processed; the basic unit of work on the System/38.
      - Routing Step – the process of initiating a job.

WORK MANAGEMENT OVERVIEW



1, A job executes under the control of a subsystem.

2, A subsystem provides a controlled environment for the execution of jobs.

3, The subsystem description defines what jobs can execute within the subsystem and how they can execute.

4, Each job must have a job description. The work entries identify the job description that has a set of attributes for the job such as the library list, routing data, and user profile.

5. Routing entries specify the class, which defines the execution priority for the job.

-The class is an object that contains execution parameters for a job such as execution priority. This priority is used when competing with other jobs for the system resources.

Progress Check – Unit 1: Work Management

Read each question carefully. Record your answers in the space provided.

1. Where is work performed in the System/38?

Work is identified to the system as a job. All jobs execute in subsystems.

2. What is a storage pool? in subsystem descriptions.

A storage pool is a logical grouping of main storage. It is an area of main storage where your job runs. You can control the number and size with a CPF command.

3. List and describe the functions of the 5 system-supplied subsystem.

QCTL - controlling subsystem - supports interactive jobs processed through the system console. automatically

started with CPF.

QINTER - interactive subsystem - supports interactive jobs processed through work stations.

QBATCH - batch subsystem supports batch jobs processed from job queues.

QSPL - spooling subsystem supports input spooling jobs processed from readers and output spooling jobs processed from output queues to writers.

QPGMR - programmer subsystem - is available to programmers for online programming.

4. Describe the 3 steps required to IMPL and start CPF.

   1. Set both rotary switches to the NORMAL positions and press the POWER ON button.
   2. Enter password
   3. Enter date and time to the Start CPF prompt.

5. Code the command to perform a controlled power down of the system with no time limit.

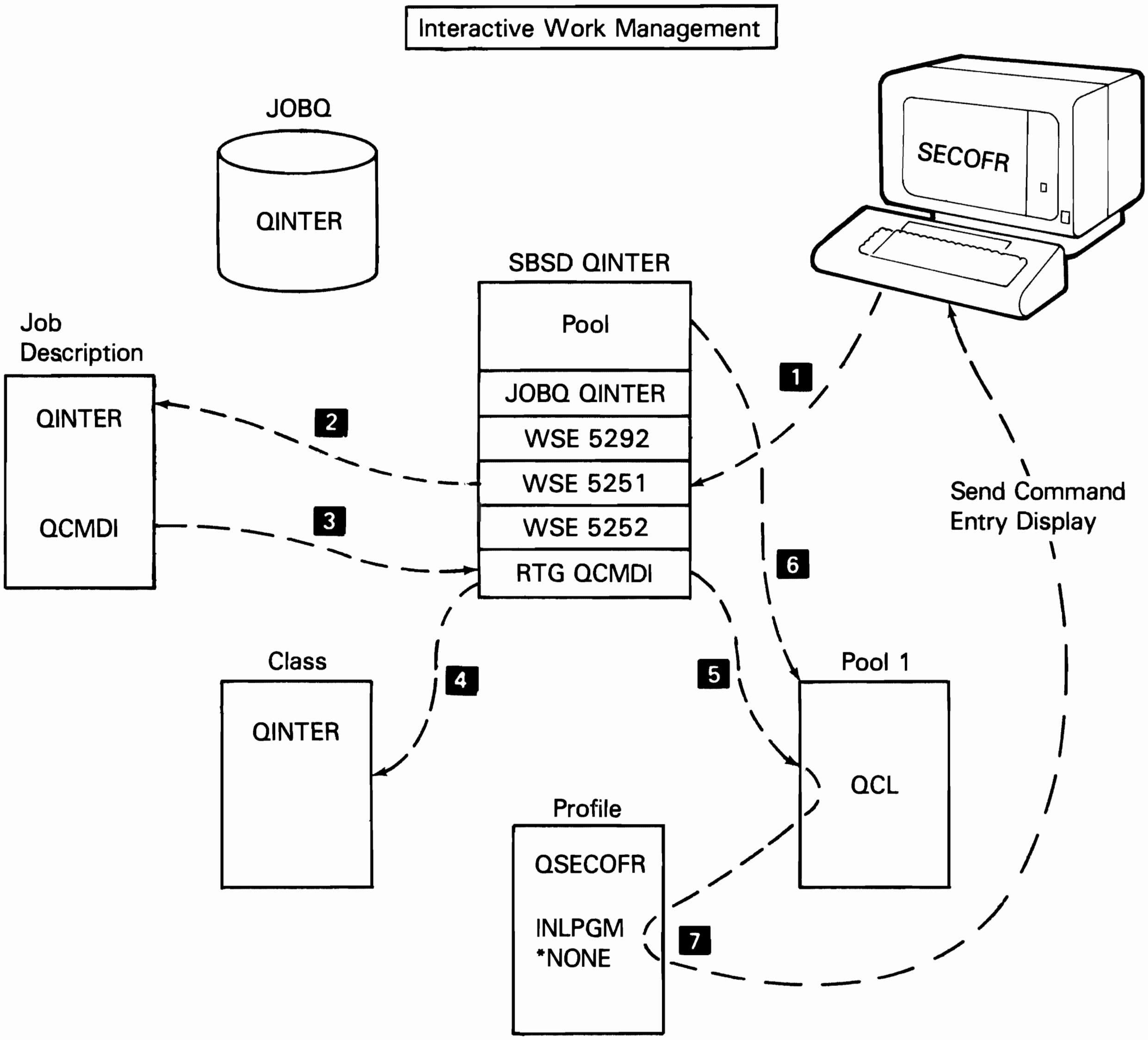
   PWRDWNSYS

   or

   PWRDWNSYS OPTION(*CUTRLD)
   DELAY(*NOLIMIT)

When you are finished, return to the module and review your answers with those in the text.

INTERACTIVE JOB SUMMARY



1 Sign-on at a work station.

1 The subsystem description must have an entry to support that work station.

2 3 The work station entry points to a job description which in turn points back to a routing entry in subsystem description.

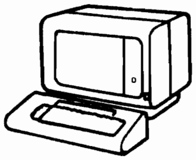
4 5 6 The routing entry points to a class, storage pool, and the name of the processing program for the routing step.

7 The processing program is QCL which checks the user profile for an initial program.

7 In this example, when the security officer signs on, the command entry screen is displayed.

To cancel a function, press the System Request SYS Req ATTN and enter enter message keys to obtain the System Request Menu

Machine Exercise – Unit 2: Interactive Job Processing



The purpose of this exercise is to show you how to:

- Create and use an output queue.
- Display and change your interactive job.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

Note: Substitute your initials for **xxx** in the instructions.

1. Sign on your work station with the course password, S38IMPL.

2. When the System/38 Implementation Topics Course Menu appears, select option 1 and enter your initials. This option creates your library and any source files you may need.

3. When the Programmer Menu appears, create an output queue named **OUTQxxx** in your library **ITLIBxxx**. Allow other users to display any file in your output queue.

4. Enter the CL command to display your interactive job.

5. Select option 3 from the Display Job Menu to display your job definition. The Job Definition Attributes display appears.

   Note your output queue name. It should be **GLCOUTQ**.

6. Press CF 1 to return to the Programmer Menu.

7. Enter the CHGJOB command to change your interactive job to use your output queue, **OUTQxxx**, in your library.

8. Repeat Steps 4 and 5.

   Again, note your output queue name. It should now be **OUTQxxx** in your library. If it isn't, repeat Step 7.

Read chapter on Performance Tuning in IBM System/38 CPF Programmer's Guide.

The CHGJOB command can change several attributes of your job, such as priorities, date, switches, and timeslice.

9. Press the Enter key to return to the Display Job Menu.

10. You may now select any or all options to review your job attributes. When finished, press CF 1 to return to the Programmer Menu.

11. Select the option from the Programmer Menu to edit a Control Language program named **STARTxxx**. The source member already exists in the QCLSRC source file in your library.

12. Review the source program. When finished, press CF 1 to exit SEU. You do not need to type or change any entries for the source program.

13. When the SEU EXIT menu appears, enter a 'Y' in the ''Print source listing'' field. Press the Enter key.

14. Enter the DSPOUTQ command to display your output queue.

    You can identify your file by your job name.

15. Select option 1 on the Output Queue display to list your spooled file.

    You can further identify your source listing by the member name, **STARTxxx**.

16. Press the Enter key to return to the Output Queue display.

17. On the Output Queue display, select option 3 for your file.

18. Locate the OUTQ parameter on the Change Spooled File Attributes Prompt. Change the value to **GLCOUTQ**. Next, change the library name for the OUTQ parameter to *LIBL. GLCOUTQ is in the QGPL library and this name is in the user library list. Then press the Enter key.

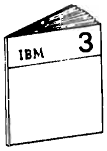
19. Press CMD 5 to redisplay when the Output Queue display appears. You should find that your file has been removed from your queue.

20. Press CF 1 to return to the Programmer Menu.

21. Press CF 1 to return to the Course Menu.

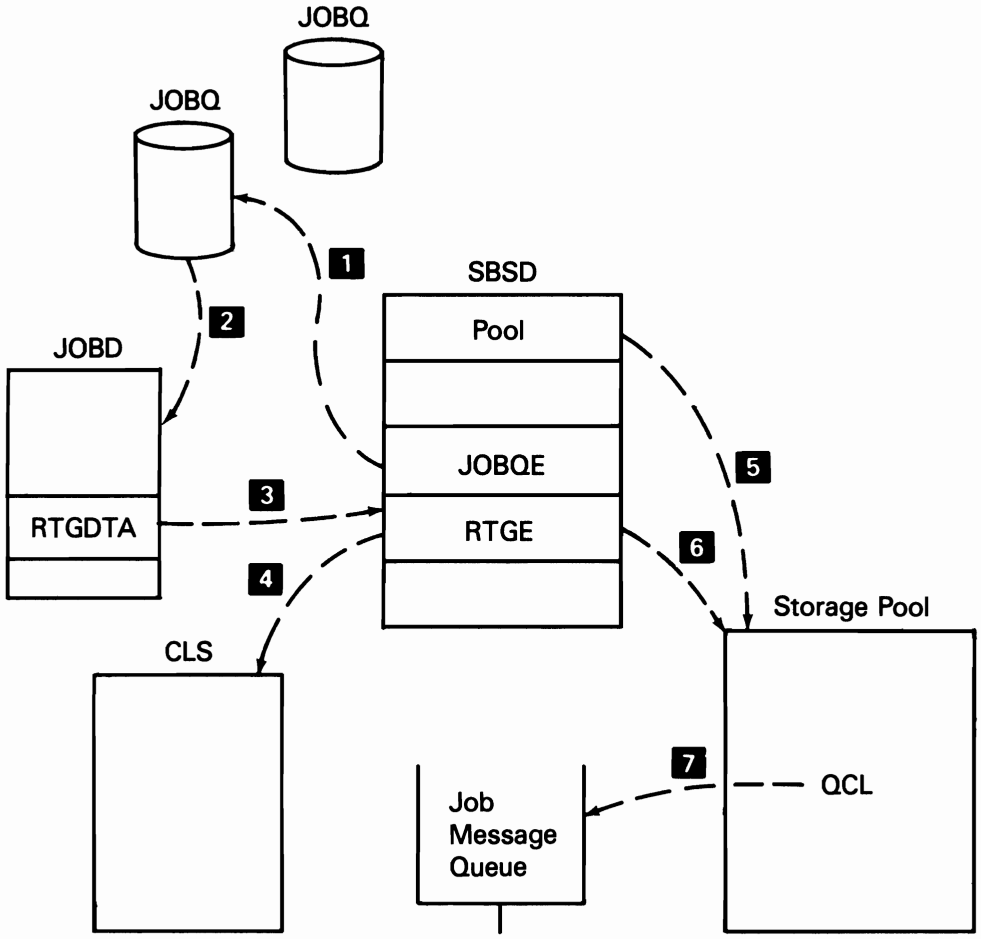
22. Select option 90 to sign off. You have now completed the steps for this exercise.

23. Remove your program listing from the work station printer.



When you are ready, return to your module text and begin with Unit 3 — Batch Job Processing.

BATCH JOB SUMMARY



The following steps occur in starting a batch job:

1 The subsystem description must have a job queue entry that points to a job queue. The job is selected from the job queue that has the name of the . . .

2 Job description which contains routing data that is used to select a . . .

3 Routing entry in the subsystem description. The routing entry points to a . . .

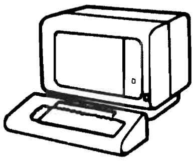
4 Class, . . .

5 Storage pool, . . .

6 and the processing program for the routing step. The processing program is QCL which checks the . . .

7 Job message queue for commands to execute.

Machine Exercise – Unit 3: Batch Job Processing



The purpose of this exercise is for you to:

- Use an output queue.
- Display and change your batch job.
- Create a job description.
- Submit a job to the batch subsystem.
- Run a job stream.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

Note: Substitute your initials for **xxx** in the instructions.

1. Sign on to your work station with the course password, S38IMPL.

2. When the System/38 Implementation Topics Course Menu appears, select option 2 and enter your initials to display the Programmer Menu. Your library and files should still exist from the previous machine exercise.

3. Create a job description named **JOBDxxx** and store it in **your** library. Your job description will have the following parameters:

```
INLLIBL = (ITLIBxxx GLCITLIB QGPL QIDU QTEMP)
OUTQ = OUTQxxx
```

   Take the default values for the other parameters.

4. Enter the command to display your job description. Verify that it was created in **your** library with the requirements stated in the previous step.

5. Press the Enter key once to see Routing and Request data values. Press the Enter key again to see the library list. Verify that it is correct.

6. Press the Enter key to return to the Programmer Menu.

7. Change the job description (Jobd:) value on your Programmer Menu to one you just created, **JOBDxxx**.

8. Use option 3 of the Programmer Menu to create a control language program named **LIBLIST**. The LIBLIST source program already exists in the QCLSRC file in your library, ITLIBxxx.

9. Display your output queue, OUTQxxx. Your compiled program listing should be in this output queue as a result of using your job description. LIBLIST should appear under the Job Name heading.

10. Select option 1 of the Output Queue display to view the program. This program replaces the library list and changes the job's output queue.

11. Press the Enter key to return to the Output Queue display.

12. Select the option to cancel the program listing from your output queue. Press the Enter key.

13. Press CF 1 to return to the Programmer Menu.

14. Enter the command to create a printer device file named **APCHECKS** in **your** library. This file is to use output queue **OUTQxxx**. Take the default for the other parameters.

    Note: Program APR150, that you will run in a later step, uses APCHECKS as the device file for printed output.

15. Enter the command to display the file description of your printer device file, APCHECKS.

16. Press the ROLL ↑ ROLL↑ ↑ key twice to verify that your output queue name is correct, OUTQxxx.

17. Press the Enter key to return to the Programmer Menu.

18. Submit (SBMJOB command) a job named **JOBxxx** to the QBATCH subsystem from your work station with request data of **'CALL APR150'**. Include your job description, **JOBDxxx**, on the submit job command. Hold your job on the QBATCH job queue.

19. Enter the command to display the QBATCH job queue.

20. Select the option to release your job. Your job executes the APR150 program that produces accounts payable checks.

21. Press the Enter key to return to the Programmer Menu.

22. Enter the command to display your output queue. Locate the spool file, APCHECKS, resulting from your submit job request.

23. Select option 1 to view your file.

24. Press CF 1 to return to the Programmer Menu.

25. Enter the command to clear **your** output queue.

26. Enter the command that starts a data base reader to read a job stream that already exists in the course library with the following parameters:

    - FILE: COPYFILE.GLCITLIB
    - MBR: COPYJOB
    - MSGQ: *REQUESTER

    The job prints the first eight records from the APPVEND data base file.

    The commands in the job stream are shown below.

```
//JOB  JOB(CPYJOB) OUTQ(GLCOUTQ)
CPYF  FROMFILE(APPVEND.GLCITLIB) TOFILE(*LIST) TORCD(8)
//ENDJOB
```

27. At the Programmer Menu, press CF 6 to display messages.

28. Press CF 8 to remove your messages and to return to the Programmer Menu.

29. Press CF 1 to return to the Course Menu.

30. Select option 4 and enter your initials to remove your course materials.

31. Select option 90 from the Course Menu to sign off the display station.

32. Then, remove your listing from the Guided Learning Center printer and return to your study carrel.

When you are ready, return to your module text and begin with Unit 4 — Work Management Considerations.

For a batch job stream, the job message queue contains all the commands between the //JOB and the //ENDJOB statements.

When you place a job into a job queue, only job related information goes in the job queue. The actual individual commands to execute are placed in the job message queue. QCL retrieves these commands from the job message queue after determining which job in the job queue should be run.

Module Exercise – Module 3: Work Management

Read each question carefully. Record your answers in the space provided.

1. Where are jobs processed on the System/38?

   Jobs are processed within a subsystem.

2. When option 6 on the programmer menu is used to submit a job, to what subsystem is it submitted?

   The job is submitted to the job queue described in the job description.

3. List two ways to start a batch job from the interactive subsystem.

   a. Submit Job (SBMJOB) command.

   b. Start Diskette Reader (STRDKTRDR),
   Start Card Reader (STRCRDRDR)
   Start Data Base Reader STR DB RDR

4. What is the function of a job queue?

   An object on which batch jobs are placed when they are submitted to the subsystem and from which they are selected for processing.

5. Can jobs be placed in the job queue without the spool subsystem being active?

   Jobs can be placed in the job queues with the SBMJOB, SBM DBJOB, SBMDKTJOB, and SBM CRD JOB commands. These commands do not require the spooling subsystem.

6. Does spool support tape devices?

   No

When you are finished, return to the module and review your answers with those in the text.

Module Summary

A job is the basic unit of work on the System/38. Work Management allocates resources for use with jobs, manages jobs on the system, and schedules jobs for execution. The Control Program Facility provides all the functions necessary for a complete working system. You can change these functions where desired through control language commands to meet your application requirements.

A job executes under the control of a subsystem. A subsystem provides a controlled environment for the execution of jobs.

System/38 provides five subsystems:

- A controlling subsystem
- An interactive subsystem
- A batch subsystem
- A spooling subsystem
- A programmer subsystem

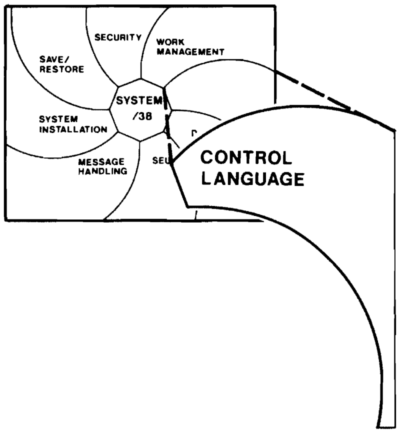
In addition to the supplied subsystems, you can define your own subsystems. You can also use control language commands to tailor a subsystem to meet your needs.

At this time, continue your course of study with the Introduction to Module 4, Control Language Programming, on the next page.

The interactive and batch subsystems function essentially the same except for two differences.

1. where the jobs are initiated from
2. where the command processor (QCL) retrieves the initial command for processing.
3. The interactive subsystem, QCL checks the user profile for the initial program. For batch, QCL checks the job message queue for commands to execute.

Module 4. Control Language Programming



In the previous System/38 Application Programming course you were introduced to Control Language (CL) programs. In this and the previous course you have entered CL commands individually. In this module you will learn how to create an executable program with a series of CL commands. Used this way, Control Language is a high level language, similar in many respects to languages like COBOL and PL/I. Also, in this module you will learn about the CL commands that can only be used in a CL program.

Before starting your study of this module you need the following materials and reference manuals. Get any of this material you do not have from your Guided Learning Center Administrator at this time.

- Paper and pencil
- The Module 4 text
- The following reference manuals:
  1. IBM System/38 Control Language Reference Manual (SC21-7731)
  2. IBM System/38 Control Program Facility Programmer's Guide (SC21-7730)
  3. IBM System/38 Source Entry Utility Reference Manual (SC21-7722)

Begin your study of Control Language programming in the Module 4 text now.

STUDENT NOTES: Module 4. Control Language Programming

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

Free form means you can start coding in any of the 80 positions of a coding line and you do not have to be concerned about extra spaces between coded elements.

The source file is where you store your CL program for compilation.

Command syntax requires that you code at least one blank between the command name and the first parameter, and between each parameter.

Keyword

START: SNDRCVF RCDFMT(MENU) — 2nd parameter listed after command name SNDRCVF

START: SNDRCVF *N MENU (positionally)

When you code parameters with keywords, the parameters can be coded in any order. When you code parameters positionally, the parameters must be in the exact order specified in the command's syntax diagram.

In order to maintain it's second position, because in this case there was no need to code a value or option for the first parameter, an *N is coded for the first parameter. *N is a special predefined value of CPF and indicates a null. You use it as many times as needed to correctly order parameters coded without keywords.

Many commands have a limit to the number of positional parameters that can be coded. The symbol ⟨P⟩ indicates the last parameter to be coded positionally. The next parameter, if coded, would require its keyword.

Within the designated positional limit you can code parameters in both forms.

Coding parameters with Keywords is recommended for CL Programming.

Continuation the + and - signs

When you use -, any blanks on the succeeding line, preceding the first non-blank character, are included in the compiler processing.

A command you enter on the Command Entry display is being entered into a single record. Therefore, no matter how many lines you use to enter a command on the Command Entry display, you enter it as if it were being entered in one continuous line.

COMMENTS. In a CL program you start a comment with /* and end with */.

You can code a comment wherever a blank can appear. Common practice, however, is to code comments on separate lines for readability as shown in the following example.

The PGM and ENDPGM commands are always coded as the first and last commands of a program.

Declare commands precede the executable commands. You can declare only one file in a CL program. This limits you to one display file or one data base file per program. Remember, a display file can contain many formats for many different displays.

Defining Variables

Every variable used in a CL program must be declared with a DCL command. The compiler automatically declares the variables that appear in the program and in the declared file

All program variables in a CL program have an & prefix

An indicator variable name in a CL program consists of the indicator number with an &IN prefix. ie the indicator 05 would appear as &IN05 in a CL program. Usually you do not have to code a DCL command for an indicator variable. All indicators are coded for you by the compiler which gets them from the display file identified by the DCLF command.

An ELSE command is optionally used in conjunction with each IF command in order to compile more efficient executable code.

Relational operators are available to handle comparisons *EQ *GT *LT *GE *LE *NE *NG *NL

You use logical operators to resolve one or more true-false logical values in a complex logical expression into one logical value. Remember, a logical expression no matter how complex is evaluated to a single true or false. In the following example, both logical expressions or conditions must be true before program APPG32 is called

```
IF COND(&A = 1) then (IF COND(&B = 5) +
   Then (call APPG32))
```

another way
The logical operator *AND specifies that the values of both relational expressions must be true for the logical expression to be true.

IF COND((&A = 1) *AND (&B = 5)) THEN(CALL APPG32)

The logical operators with their equivalent symbols are as follows

*AND - Both true to be true
*OR - Either or both to be true
*NOT - Negates a true or false
*NOT false = true
*NOT true = false

&EOF is defined as a logical variable by specifying its type as *LGL
DCL VAR(&EOF) TYPE(*LGL)
A logical variable can have one of two values: '1' which represents true and '0' which represents false. Because of this, logical variables can be used in logical expressions of the IF command. In the following example, if &EOF has the value '1' the condition or logical expression would be true and the GOTO command executed IF COND(&EOF) THEN(GOTO CMDLBL(STMT90))

CALLING ANOTHER Program.
You use the CALL compond to pass control to another program, and optionally, to pass data between two programs. not pass data to

Progress Check – Unit 1: Using CL Programs

Read each question carefully. Record your answers in the space provided.

Which of the listed program languages can you use to do the following?

a. CL
b. Other System/38 Languages
c. Any of the above.

a 1. Create a file.

c 2. Read a data file.

b 3. Update a data file record.

c 4. Read and write a display file.

b 5. Use display subfiles.

a 6. Override print file attributes.

b 7. Print data using a printer file.

When you are finished, return to the module and review your answers with those in the text.

You cannot use CL to do the following:
1. Add or update records in a data base file. You can, however, read a record from a data base file.
2. Print records from a file using a printer file. You can, however, print record images from a file with the copy command.
3. Process communication files.
4. Process display subfiles. You can, however, use a message subfile in a CL program to display error messages.
5. Describe a record in a program as you can with RPG III input specifications and in the COBOL data division.

Nevertheless, except for the limitations mentioned above, the CL programming language is a full function language. For instance, the CL programming language has the following in common with the other System/38 languages;

- Decision logic such as IF and GOTO statements
- Arithmetical capability of add, subtract, multiply and divide
- Definition capability for variables
- Input-output capability to write and read a display file, and to read a single-format data base file sequentially.
- Usable in both batch and interactive jobs
- Inter-program communication capabilities to pass control and data between programs.

When programming an application, you use a CL program to create and delete objects such as files, data areas and libraries. Also, you use a CL program to check for the existence of objects.

You use a CL program to control job and file attributes in an application.

You may elect to use a CL program as the main controlling program of an application. Because of the unique functional capabilities of CL commands, CL programs lend themselves to this use. In an interactive application this main program controls the main menu display file and calls other programs.

As you go through this module you will study many CL programs which illustrate their typical uses.

On the other hand, if you need to update a file, print reports, use a communication file or process a display subfile, you must use the other System/38 languages.

Progress Check – Unit 2: Coding CL Programs

Read each question carefully. Record your answers in the space provided.

Indicate whether each of the following is true (T) or false (F).

F 1. If you start to code the parameters of a command positionally, you must code any following parameters positionally.

F 2. One or more blanks are required between a label and a command. The blank(s) are optional, because a colon, not a blank, delimits a label name.

T 3. ##STMT103 is a valid label name. A label name must begin with A-Z, $, # or @, which can be followed by ->

T 4. Parameters are separated by one or more blanks.

F 5. You cannot code a comment between a command and a parameter.

When you are finished, return to the module and review your answers with those in the text.

The CL command as used in a CL program has three parts which are arranged in the following order:

Label: Command-name Parameters

You may label every command in your program. In this respect, you will find labels are useful for program debugging.

-> as many as 9 characters from the set A-Z, 0-9, $, #, @, or _. Therefore, a name cannot exceed ten characters nor can it have embedded blanks.

send receive file

COMMANDS The structure of a command – Verbs object SND RCV F

Many times an adjective is used to qualify the object

CHG SYS VAL

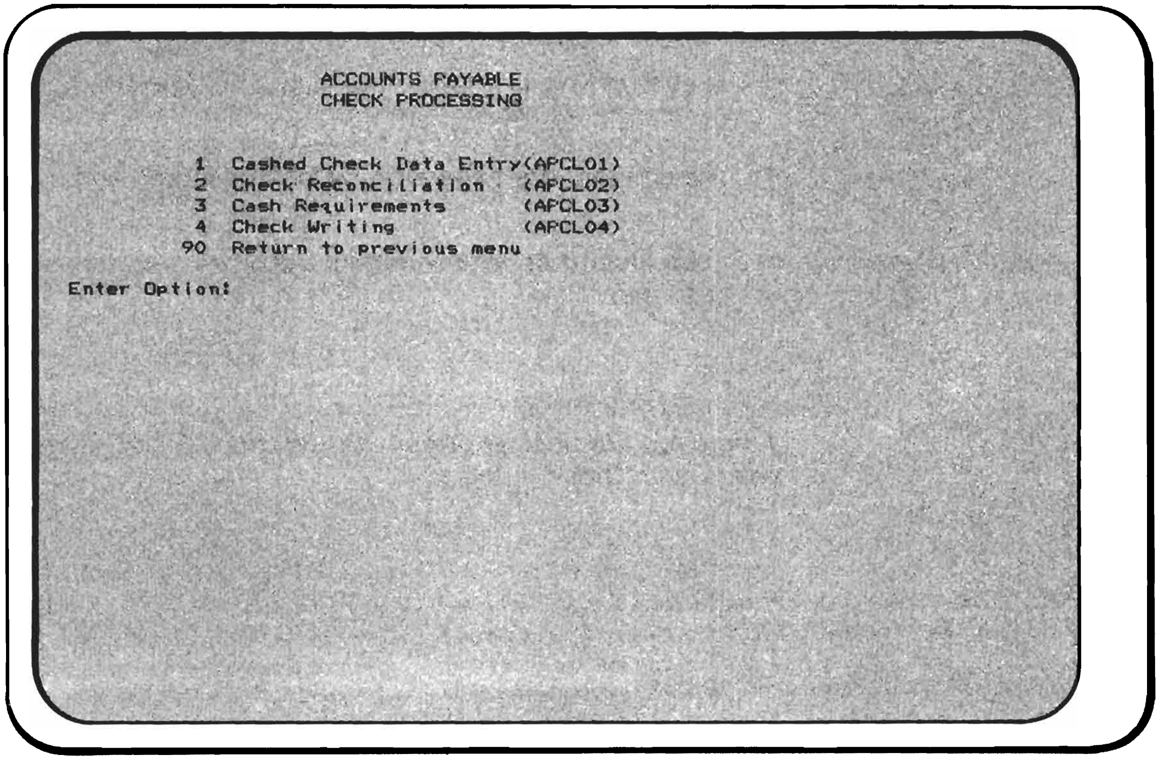
change system value

Machine Exercise – Unit 3: Coding a CL Program for a Menu Display

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

Substitute your initials for xxx in the instructions.

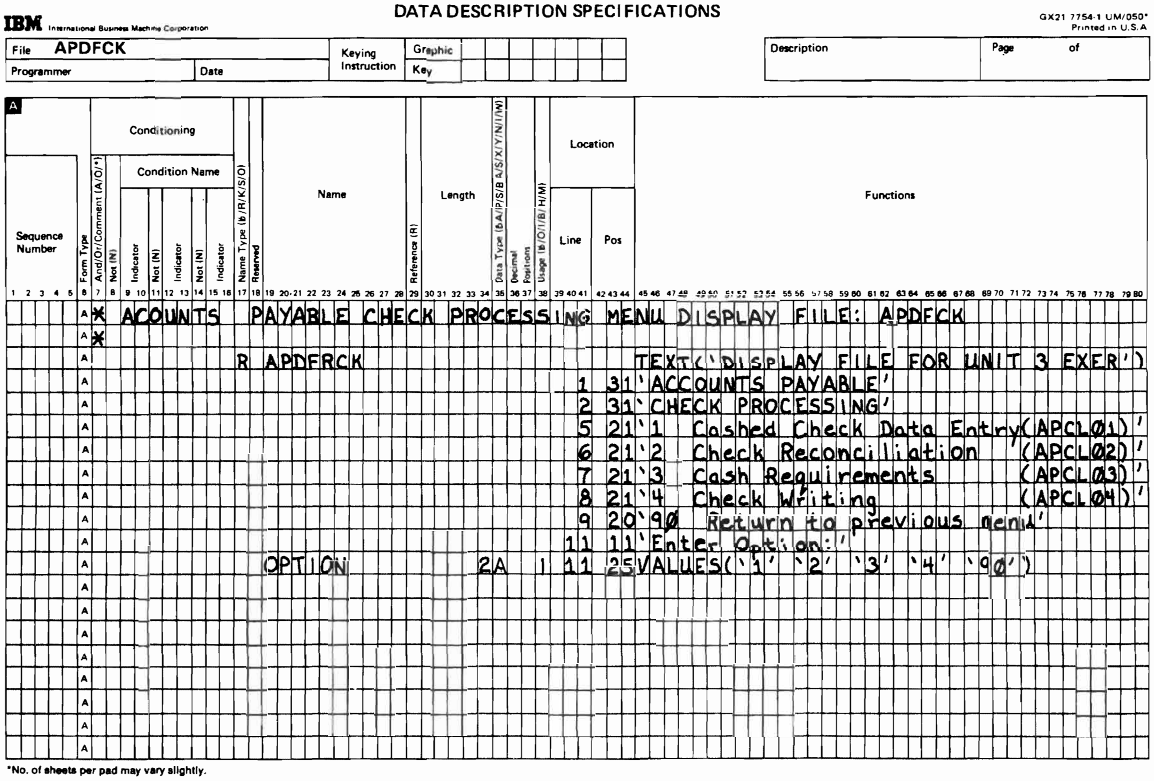
In this exercise, you are asked to code, compile, and execute a CL program to control the following menu.



```
                    ACCOUNTS PAYABLE
                    CHECK PROCESSING

              1  Cashed Check Data Entry(APCL01)
              2  Check Reconciliation    (APCL02)
              3  Cash Requirements       (APCL03)
              4  Check Writing           (APCL04)
             90  Return to previous menu

Enter Option:
```

The display file for the menu and the programs you will call from your program are in the library GLCITLIB. This library name has been put in all the required library lists for you. As a result, you do not need to qualify any file or program name in your program.

The DDS for this file, which you will need to reference for your coding, is shown below.



IBM International Business Machines Corporation

DATA DESCRIPTION SPECIFICATIONS

GX21 7754-1 UM/050*
Printed in U.S.A.

File APDFCK
Programmer Date
Keying Instruction Graphic Key
Description Page of

```
     A* ACOUNTS  PAYABLE CHECK PROCESSING MENU DISPLAY FILE: APDFCK
     A*
     A          R APDFRCK                   TEXT('DISPLAY FILE FOR UNIT 3 EXER')
     A                                   1 31'ACCOUNTS PAYABLE'
     A                                   2 31'CHECK PROCESSING'
     A                                   5 21'1  Cashed Check Data Entry(APCLØ1)'
     A                                   6 21'2  Check Reconciliation   (APCLØ2)'
     A                                   7 21'3  Cash Requirements      (APCLØ3)'
     A                                   8 21'4  Check Writing          (APCLØ4)'
     A                                   9 20'9Ø  Return to previous menu'
     A                                  11 11'Enter Option:'
     A            OPTION         2A  I 11 25VALUES('1' '2' '3' '4' '9Ø')
     A
     A
     A
     A
     A
     A
     A
     A
```

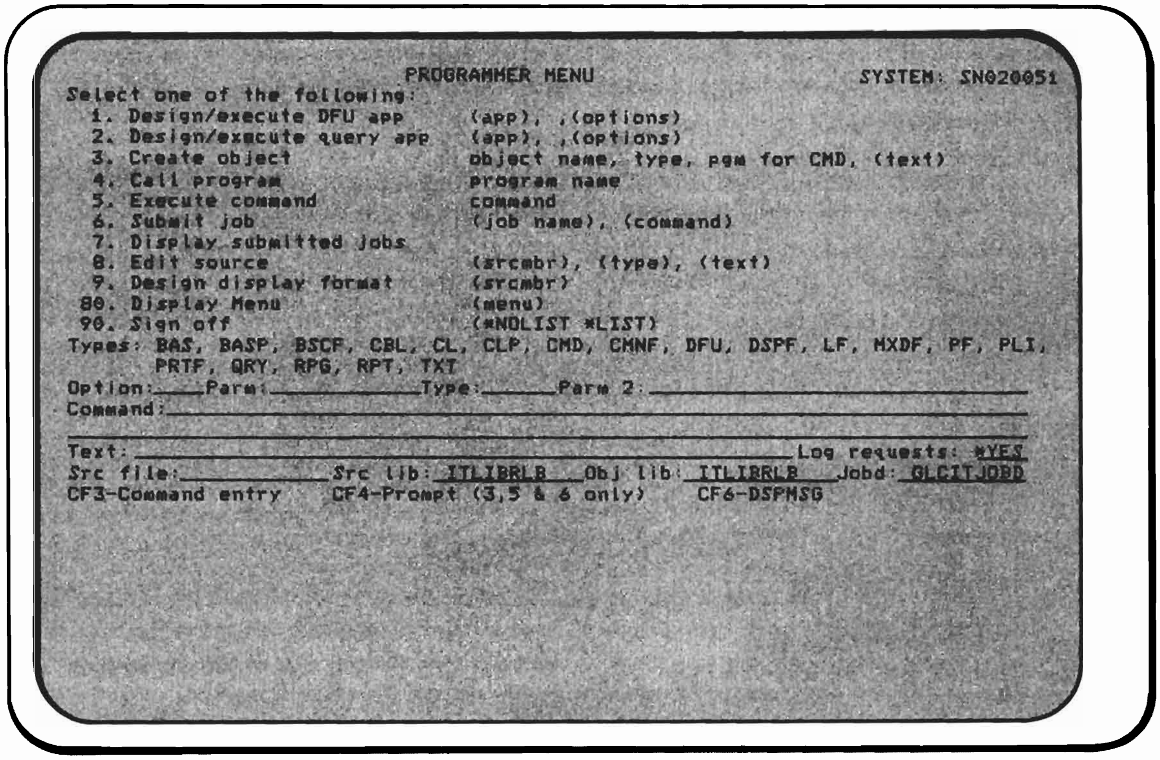
*No. of sheets per pad may vary slightly.

1. Code your program following the instructions below.
   a. Name your program APCL3xxx, where xxx are your initials.
   b. Use the IF and ELSE logic commands to determine the option entered.
   c. Code your program to call the program name shown in parenthesis on the menu when its option is entered.
   d. Exit the program with a RETURN command.
   e. Code your program on paper or in the Notes section for Module 4, here in your Student Materials Book.
   f. Use the appropriate material as the need arises.

2. When you have finished coding and checking your program, go to a display station attached to a System/38 and enter the password S38IMPL to sign on.

3. From the System/38 Implementation Topics Course Menu, select option 1 to create a library. Be sure to enter your initials.

   When this operation is complete, the Programmer Menu appears on the display. A sample Programmer Menu appears below.



```
                     PROGRAMMER MENU                        SYSTEM: SN020051
Select one of the following:
  1. Design/execute DFU app      (app), ,(options)
  2. Design/execute query app    (app), ,(options)
  3. Create object               object name, type, pgm for CMD, (text)
  4. Call program                program name
  5. Execute command             command
  6. Submit job                  (job name), (command)
  7. Display submitted jobs
  8. Edit source                 (srcmbr), (type), (text)
  9. Design display format       (srcmbr)
 80. Display Menu                (menu)
 90. Sign off                    (*NOLIST *LIST)
Types: BAS, BASP, BSCF, CBL, CL, CLP, CMD, CMNF, DFU, DSPF, LF, MXDF, PF, PLI,
       PRTF, QRY, RPG, RPT, TXT
Option:____Parm:____________Type:______Parm 2:____________________________
Command:__________________________________________________________________
__________________________________________________________________________
Text:__________________________________________________Log requests: *YES
Src file:___________Src lib: ITLIBRLB   Obj lib: ITLIBRLB    Jobd: GLCITJOBD
CF3-Command entry    CF4-Prompt (3,5 & 6 only)    CF6-DSPMSG
```

4. Select option 8, Edit source, to use SEU to enter your program into a source member. The Type is CLP for CL program. You may want to have the IBM System/38 Source Entry Utility reference manual and User's Guide (SC21-7722) available. It is located in the Guided Learning Center Resource Library.

5. Enter your source statements.

   When entering command statements you will find it very helpful to enter the command and then press CF 4. This will not only prompt for the parameter values but supply the parameter keywords and align your command statements. You can even use CF 4 within the prompt. For instance, when you code a command in the THEN parameter of an IF command, you can enter CF 4 and get the prompt display for the command you just entered.

   Use CF 1 to end SEU.

6. When you have finished entering your program, compile it by entering option 3 from the Programmer Menu.

7. When you have gotten a successful compile, run your program from the Programmer Menu with option 4.

   The Accounts Payable Check Processing menu should appear on the display.

8. If the Accounts Payable Check Processing menu appears on the display, continue with Step 9.

   If the Accounts Payable Check Processing menu does not appear, review your program, make any corrections, and begin again with Step 4.

9. Test your program by entering the options 1, 2, 3, and 4. When you enter a correct option, the program called issues a message informing you that the call was successful. Follow the instructions of the message to return to the Accounts Payable Check Processing menu.

10. After you have tested each option, select option 90 to return to the Programmer Menu.

    If your program processed propertly, continue with Step 11.

    If your program did not run correctly, determine the error, and begin with Step 4 again.

11. With the Programmer Menu displayed, press CF 6 to display your messages. Each of the messages you saw earlier should appear on the display.

12. Press CF 8 to return to the Programmer Menu. This deletes all the messages you saw.

13. Press CF 1 to return to the Course Menu.

14. Select option 90 from the Course Menu to sign off the display.

Your Administrator has a copy of a possible solution for the CL program. You may get a copy if you feel you need it and it will be of help to you.

When you are finished with the exercise, return to Unit 4 of the Module 4 text.

Progress Check – Unit 4: Communicating Between Programs

Read each question carefully. Record your answers in the space provided.

Indicate whether each of the following statements is true (T) or false (F).

T 1. You can pass data with a transfer control (TFRCTL) command.

T 2. You can code a variable for the program name in a CALL command.

T 3. You can code a variable for the program name in a TFRCTL command.

F 4. The PGM command in the calling program lists the parameters to be passed.

F 5. You can code a constant as a parameter in a TFRCTL command.

T 6. Decimal constants are always passed with length attribute of (15 5).

F 7. Character constants are always passed with a length of 32 bytes.

F 8. The following coding example passes data correctly.

```
PGM  PARM(&A &B &C)
DCL VAR(&A) TYPE(*DEC) LEN(8 2)
DCL VAR(&B) TYPE(*CHAR) LEN(10)
DCL VAR(&C) TYPE(*CHAR) LEN(20)
DCL VAR(&D) TYPE(*DEC) LEN(5)
       :
       .
TFRCTL PGM(APVND) PARM(&A &D)
ENDPGM
```

T 9. The program at the bottom of the invocation stack is the program that is executing or has control in a job.

F 10. The calling program can address the data in the parameter passing storage area.

When you are finished, return to the module and review your answers with those in the text.

4. The PGM command in the called program lists the parameters passed to it and this list must agree in number, order and data attributes with the list in CALL command.

5 Coding a constant introduces new data into parameter list, which you cannot do with a TFRCTL command. TFRCTL can only pass data which was passed to its program.

7 32 characters is the minimum length that is passed. If the character string is longer than 32 characters, you must specify the number of characters in the string in the DCL of the variable to contain the constant in the called program.

8. The parameter list contains the variable &D which introduce new data into a list.

10 Only programs to which control is passed address the data in the data passing storage area. The calling program can manipulate the variables in the CALL PARM list. The storage area is loaded with whatever value is in the variable when the CALL command is executed.

Progress Check – Unit 5: Creating Objects with a CL Program

Read each question carefully. Record your answers in the space provided.

1. List the three types of data you can define with the DCL command.

   Decimal (*DEC)
   Character (*CHAR)
   Logical (*LGL)

2. Write a CHGVAR command to multiply &A and &B and round the answer, &PROD, to the nearest cent.

```
DCL VAR(&A) TYPE(*DEC) LEN(5 2)
DCL VAR(&B) TYPE(*DEC) LEN(5 4)
DCL VAR(&PROD) TYPE(*DEC) LEN(10 2)
```

   CHGVAR VAR(&PROD) VALUE((&A *&B) + .005)

3. Given the following data values

```
&A = 18.40
&B = 14
&D = 3.2
```

   what is the value of &ANS when the following CHGVAR command is executed?

```
DCL VAR(&A) TYPE(*DEC) LEN(4 2)
DCL VAR(&B) TYPE(*DEC) LEN(5)
DCL VAR(&D) TYPE(*DEC) LEN(5 2)
DCL VAR(&ANS) TYPE(*DEC) LEN(10 1)
   :
   .
CHGVAR VAR(&ANS) VALUE(((&A * &B)/&D) + .05)
```

   80.5

4. Given

```
&X = '1'
&Y = '1'
&Z = '0'
```

is &IN01 true or false when the following command is executed?

```
CHGVAR VAR(&IN01) VALUE(&X *AND &Y) *OR &Z)
```

True

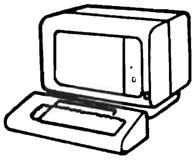
5. What is the contents of &A when the CHGVAR command is executed? &B contains 38429.

```
DCL VAR(&A) TYPE(*CHAR) LEN(6)
DCL VAR(&B) TYPE(*DEC) LEN(5 2)
   :
   .
CHGVAR VAR(&A) VALUE(&B)
```

384.29

When you are finished, return to the module and review your answers with those in the text.

Machine Exercise – Unit 6: Substring Coding



Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

Substitute your initials for xxx in the instructions.

In this machine exercise you are to code, compile and execute a CL program. The program is to use the substring function to add the digits in the character string of numbers, '820953'. Name your program SSTCLxxx.

1. Code your program observing the following items:
   a. Declare the character string as follows:

```
DCL VAR(&NBR) TYPE(*CHAR) LEN(6) VALUE('820953')
```

   b. Declare the variable to contain the sum as follows:

```
DCL VAR(&SUM) TYPE(*DEC) LEN(2)
```

   c. To display your answer, write your program to call program DSPANS with &SUM as the parameter.

2. When you have finished coding and checking your program, go to a work station and enter the password S38IMPL to sign on.

   This displays the System/38 Implementation Topics Course Menu.

3. Call the Programmer Menu by entering option 2. Remember to enter your initials.

4. Select option 8, Edit source, to use SEU to enter your program into a source member.

   When entering command statements you will find it very helpful to enter the command and then press CF 4. This will not only prompt for the parameter values but supply the parameter keywords and align your command statements. You can even use CF 4 within the prompt. For instance, when you code a command in the THEN parameter of an IF command, you can enter CF 4 and get the prompt display for the command you just entered.

   End SEU activity with CF 1.

5. When you have finished entering your program, compile it by entering option 3 from the Programmer Menu.

6. When you have gotten a successful compile, execute your program by selecting option 4 from the Programmer Menu.

7. A message displays your answer and the correct answer. Press the Enter key to end the display.

   If your answer is incorrect, review and correct your code and begin again with Step 4.

8. When your answer is the correct one and the Programmer Menu is displayed, press CF 6 to display your message(s).

9. Press CF 8 to return to the Programmer Menu.

10. Press CF 1 to return to the Course Menu.

11. Select option 4 from the Course Menu to delete your material. Remember to enter your initials.

12. Select option 90 from the Course Menu to sign off the work station.

Your Administrator has a copy of a possible solution for the CL program. You may get a copy if you feel you need it and it will be of help to you.

When you have completed this machine exercise, return to the Module 4 text and continue your study in Unit 6 at the section titled ''Character String Operators''.

Progress Check – Unit 6: Character String Operators

Read each question carefully. Record your answers in the space provided.

1. Indicate which character string operator best fits each of the following situation?

```
*CAT
*BCAT
*TCAT
```

*BCAT a. Obtain proper spacing for variables used in a message text.

*TCAT b. Build a qualified name from variables in a program.

*CAT c. Accept existing spacing of character strings.

2. Is the text in the following MSG parameter correctly coded? If not, why is it incorrect?

```
DCL VAR(&SUM) TYPE(*DEC) LEN(5 2)
   :
   .
SNDBRKMSG MSG('Your answer is' *BCAT &SUM *CAT '.')
```

Only character strings can be used with Character string operators. The variable &SUM contains decimal type data. To use the value in &SUM you would first have to convert it to character type data

When you are finished, return to the module and review your answers with those in the text.

with a CHGVAR command, as follows:

CHGVAR VAR(&CSUM) VALUE(&SUM)

Progress Check – Unit 7: Using Data Areas

Read each question carefully. Record your answers in the space provided.

Indicate the letter of the item which matches the numbered statements.

a. Retrieve data area command, RTVDTAARA
b. Local data area
c. Permanent data area
d. Declare data area command, DCLDTAARA

b 1. Created by CPF.

d 2. Creates a CL variable using the data area name.

c 3. Created by the user.

b 4. Lasts for the duration of a job.

a 5. The data area need not exist to compile this command.

d 6. The data area must exist to compile this command.

a 7. Command contains substring capability.

b 8. Can only contain character type data.

a 9. Can perform same function with CHGVAR command.

10. Code a command to create a data area named PRJTTL in the library named PRJLIB. The data area is to contain 30 characters. Other users can have normal use of the data area.

CRTDTAARA DTAARA(PRJTTL.PRJLIB)
TYPE(*CHAR) LEN(30) PUBAUT(*NORMAL)

11. Code a command to extract 10 characters starting at position 31 of the local data area. Put the data in the variable &ACCTCD.

RTVDTAARA DTAARA(*LDA (31 10)) RTNVAR(&ACCTCD)
or
CHGVAR VAR(&ACCTCD) VALUE(%SST(*LDA 31 10))

When you are finished, return to the module and review your answers with those in the text.

Desk Coding Exercise – Unit 8: Using the %SUBSTRING Function

In this desk exercise you are to recode the example in the student text in Unit 8 under the heading Using System Values assuming the system date is in julian format.

Hint: use %SUBSTRING to extract the month and day from the converted date. You are encouraged to use reference material when the need arises.

```
PGM

DCL     VAR(&DATE)  TYPE(*CHAR)  LEN(5)
DCL     VAR(&CDATE) TYPE(*CHAR)  LEN(6)
DCL     VAR(&IN02)  TYPE(*LGL)
 /* Retrieve the julian system date */
RTVSYSVAL  SYSVAL(QDATE)  RTNVAR(&DATE)
/* Convert the date format to MDY */
CVTDAT  DATE(&DATE) TOVAR(&CDATE) FROMFMT(*JUL) +
   TOFMT(*MDY) TOSEP(*NONE)
 /* TEST for leap year */
IF    COND(%SST(&CDATE 1 4) = '0229' +
  THEN(CHGVAR VAR(&IN02) VALUE('1'))
CALL  PGM(PAYCALC) PARM(&IN02)
```

When you are finished, return to the module and review your solution with the one in the text.

Progress Check – Unit 8: Getting System Values and Converting a Date Format

Read each question carefully. Record your answers in the space provided.

1. Assume a date entered from the console is to be used in a CL program to change the system value QDATE. QDATE is in the julian format. The date is entered in the format MM/DD/YY into an 8 character field named &DATE. You are to code the following three commands:

   A DCL command to declare the variable &JULDT to contain the julian date.

   A CVTDAT command to convert the date entered to the julian format.

   A CHGSYSVAL command to change the system value to the date entered.

   Do not hesitate to use the reference manuals.

```
DCL   VAR(&JULDT) TYPE(*CHAR) LEN(5)
CVTDAT DATE(&DATE) TOVAR(&JULDT) +
 FROMFMT(*MDY)   TOFMT(*JUL) TOSEP(*NONE)
CHGSYSVAL   SYSVAL(QDATE) VALUE(&JULDT)
```

Indicate whether each of the following statements is true (T) or false (F).

F 2. You can change all the system values. Some system values like the abnormal termination switch, QABNORMSW, are controlled and can only be changed by CPF.

T 3. Time system values are stored as character type data.

F 4. You can create new system values.

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 8: Job Attributes

Read each question carefully. Record your answers in the space provided.

Indicate whether each of the following statements is true (T) or false (F).

F 1. Assume the job switches are set as follows 01010001. Control will be transfered to STMT10.

```
IF COND(X1XX0XX0) THEN(GOTO STMT10)
```

T 2. Every running job has as associated set of job attributes that can be accessed by the user.

T 3. You can retrieve more than one job attribute with one RTVJOBA command.

You retrieve as many job attributes as you specify parameters for on the RTVJOBA command.

When you are finished, return to the module and review your answers with those in the text.

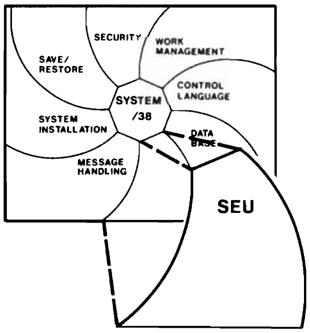
Module 4 Summary – Control Language Programming

Most applications you will develop or maintain will probably have one or more CL programs. You have learned a great deal about control language programming in this module. Following is a summary of the highlights of what you studied.

- The use of CL programs in applications.
- The syntax of coding of CL commands in CL programs.
- How to write a CL program to control a menu display, and how you use IF and ELSE commands for logic.
- The coding of the CHGVAR command to manipulate character, decimal and logical values.
- How communication is established between programs and the use of the CALL and TFRCTL commands to pass control and data between programs.
- How to use the create commands with the monitor message command to create an object in a CL program.
- The use of the built-in functions and character string operators.
- The creation and accessing with allocating of data areas in CL programs.
- How to retrieve system values and obtain job status in a CL program.

This completes your study of Control Language Programming. Continue with the next module, Source Entry Utility Additional Topics, by reading its Introduction on the next page.

Module 5. Source Entry Utility Additional Topics



Source Entry Utility (SEU) is one of the programs provided by the System/38 Interactive Data Base Utilities (IDU). SEU performs many functions with your source members, including:

- Add new records
- Change existing records
- Move records
- Copy records
- Delete records
- Scan records
- Substitute characters in records
- Copy records from one member to another
- Provide formats for high level language specification types
- Provide prompting assistance for record entry
- Syntax checking

The process of using SEU to enter and update records is called editing. SEU can be initiated by using option 8 of the Programmer Menu or by using the Edit Source (EDTSRC) command.

SEU makes extensive use of Help text, command function (CF) keys, and function control keys. A summary of command function keys and function control keys is on the following pages, as well as in the Source Entry Utility Reference Manual.

You will use the following material for study of this module:

- Module 5 text
- IBM System/38 Source Entry Utility Reference Manual and User's Guide (SC21-7722)

Use the SEU Reference Manual to assist you with any of the machine exercises in the module.

Begin your study of SEU in the Module 5 text now.

Command Function Key Summary

Command Function Key	Name of Key	Description
CF 1	Exit	Displays the Exit display to end SEU activity
CF 2	Previous display	Presents a logically prior display (not valid on some displays)
CF 4	Prompt	Invokes CL or BASIC prompting or creates a prompting section on the display
CF 5	Services Display	Displays the Services display
CF 6	Cancel Pending Operation	Cancels an operation for which you have not entered all the line commands
CF 7	Scan/ Substitute Forward	Searches for the next occurrence of a scan string and does a substitution if one is specified
CF 8	Scan/ Substitute Backward	Searches for the previous occurrence of a scan string and does a substitution if one is specified
CF 12	Uppercase/ Lowercase	Reverses status of uppercase/lowercase indicator

Function Control Key Summary

Name of Key	Description
DEL	Deletes the character in the cursor position and shifts all characters to the right of the cursor one position to the left
DUP	Duplicates a field from the previous record into the same field of the current record
ENTER/REC ADV	Enters information you have keyed
HELP	Displays SEU help text; if cursor is on the message line, the second level text for that message is displayed
HOME	Press once to move cursor to its original position on the screen; press twice to remove any keyed data that has not been entered
INS	Insert character(s); any keying causes existing data to move to the right
PRINT	Prints the current screen display
ROLL ↑	Page forward through the displayed member; if cursor is on the message line, the next message displays
ROLL ↓	Page back through the displayed member; if the cursor is on the message line, the previous message displays

STUDENT NOTES: Module 5. Source Entry Utility Additional Topics

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

LINE COMMAND

S - Defines the contents of the skeleton line and the cursor position.

IS Inserts the skeleton line and positions the cursor.

STUDENT NOTES (continued): Source Entry Utility Additional Topics

Machine Exercise 1 – Unit 2: Using the Operations of the Service Display

In this exercise you will use the Browse/Copy operation to copy source code from an existing source member into your source member. Then, you will use the Scan/Substitute operation to make changes in your source code.

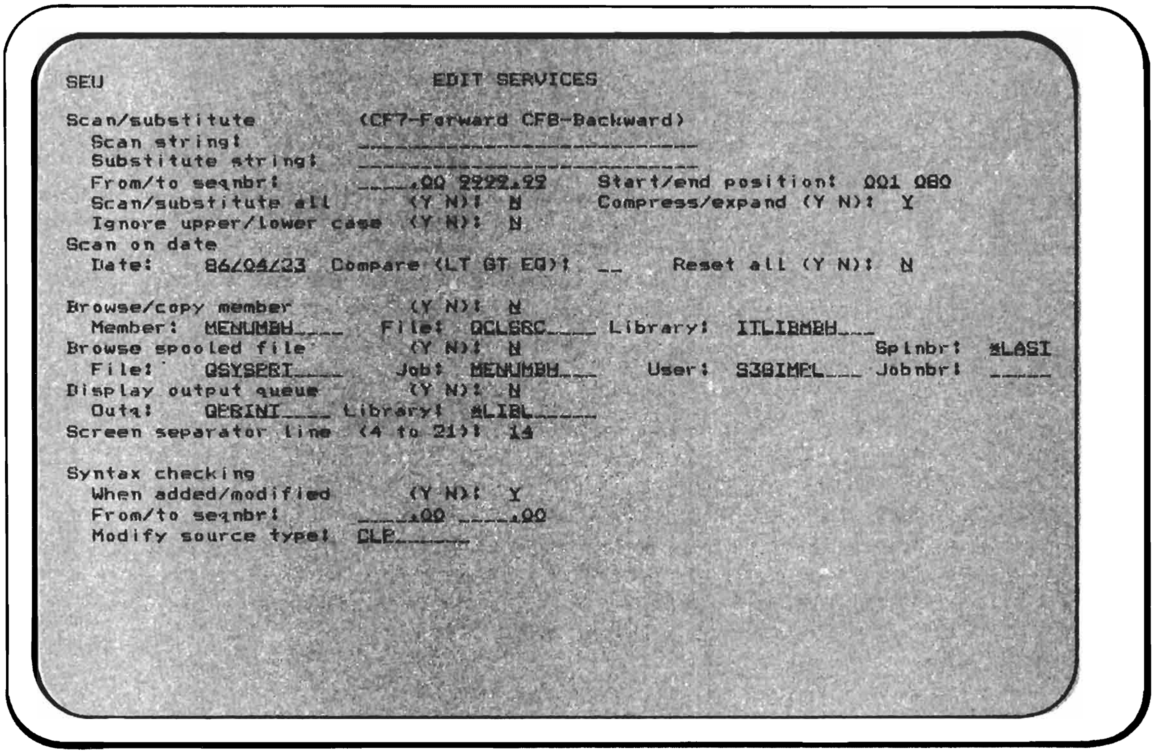
Read through the entire exercise before you go to the display station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise. Substitute your initials for xxx.

1. Sign on your work station with the course password S38IMPL.

2. Create a library by selecting option 1 and entering your initials on the System/38 Implementation Topics Course Menu.

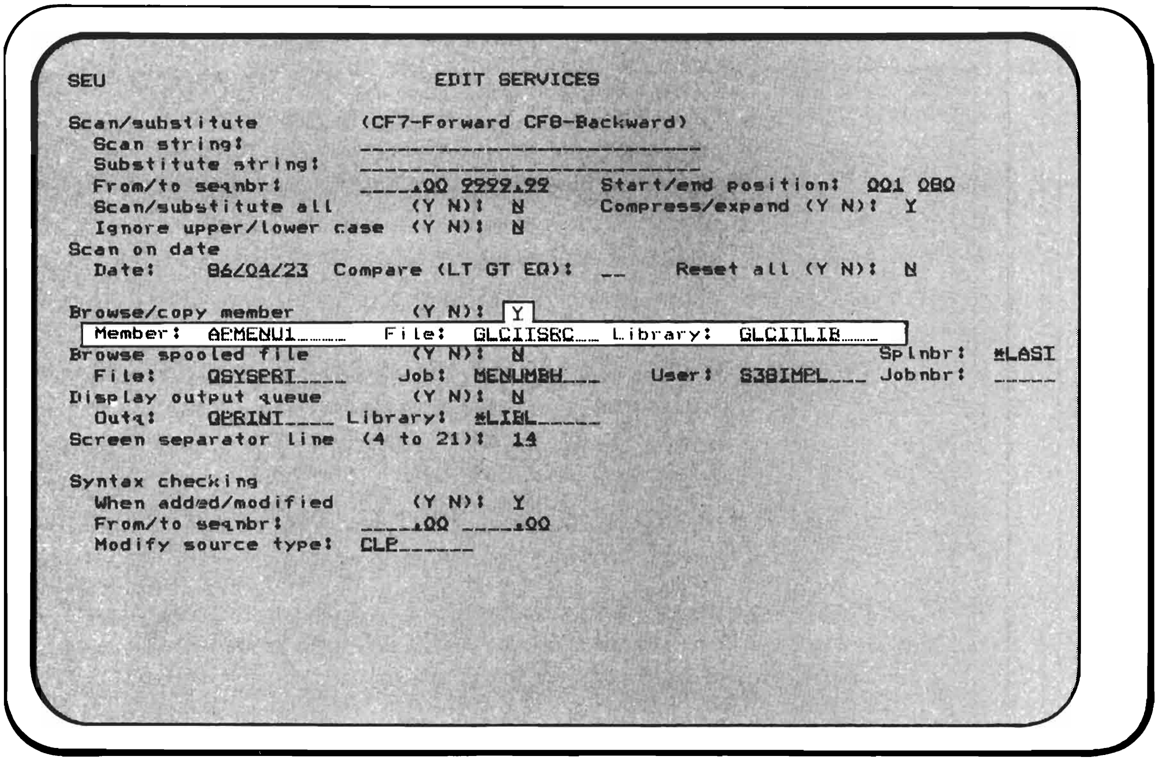
3. When the Programmer Menu appears select the Edit Source by entering option 8, program name MENUxxx and type CLP. MENUxxx is a new source member you are creating in your library.

4. When your empty source member is displayed, press CF 5 to get the Services display, similar to the one shown below.



```
SEU                                EDIT SERVICES

Scan/substitute            (CF7-Forward CF8-Backward)
  Scan string:             ___________________________
  Substitute string:       ___________________________
  From/to seqnbr:          ____.00 9999.99    Start/end position:  001 080
  Scan/substitute all          (Y N):  N      Compress/expand (Y N):  Y
  Ignore upper/lower case  (Y N):  N
Scan on date
  Date:     86/04/23  Compare (LT GT EQ):  __     Reset all (Y N):  N

Browse/copy member             (Y N):  N
  Member:  MENUMBH____    File:  QCLSRC____ Library:  ITLIBMBH___
Browse spooled file            (Y N):  N                        Splnbr:  *LAST
  File:    QSYSPRT____      Job:  MENUMBH___     User:  S38IMPL___ Jobnbr:  _____
Display output queue           (Y N):  N
  Outq:    QPRINT____  Library:  *LIBL______
Screen separator line  (4 to 21):  14

Syntax checking
  When added/modified          (Y N):  Y
  From/to seqnbr:          ____.00 ____.00
  Modify source type:  CLP______
```

5. Use the Browse/Copy operation to access the member APMENU1. It is in the file GLCITSRC in the GLCITLIB library. The completed Services display is shown below. Make your entries, then press the Enter key.



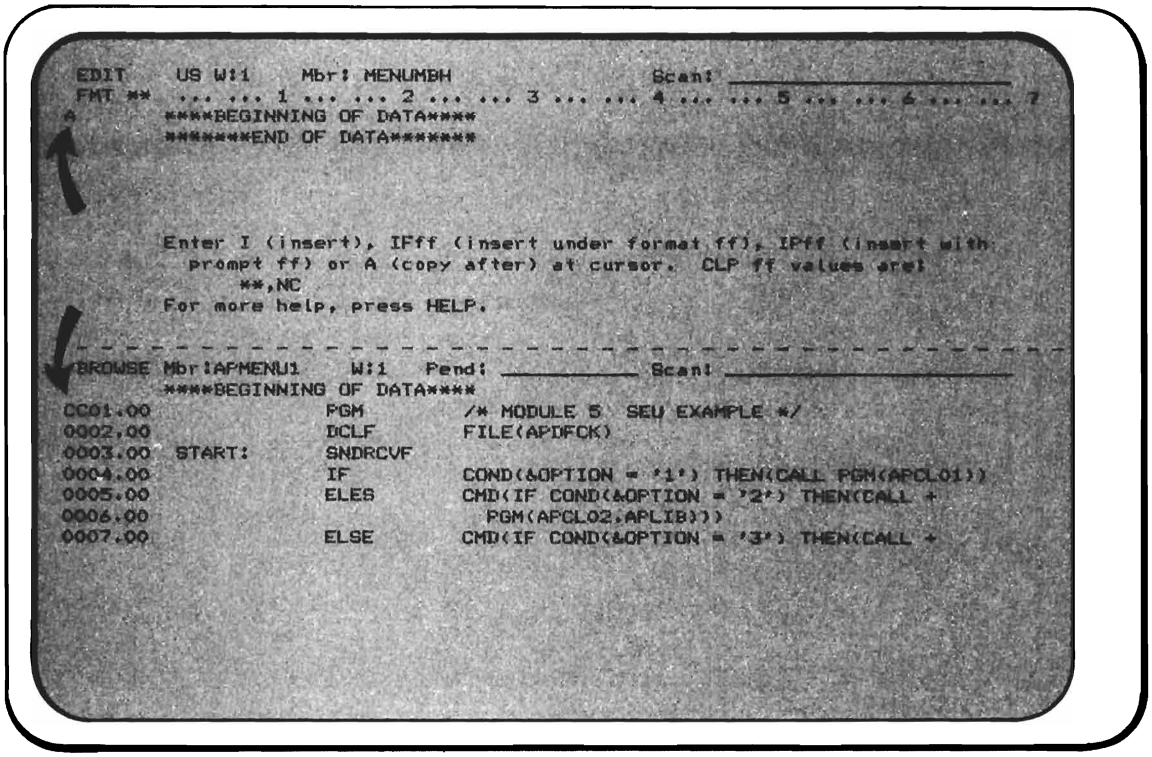
```
SEU                              EDIT SERVICES

Scan/substitute             (CF7-Forward CF8-Backward)
  Scan string:              ____________________________
  Substitute string:        ____________________________
  From/to seqnbr:           ____.00 9999.99     Start/end position:  001 080
  Scan/substitute all         (Y N):  N         Compress/expand (Y N):  Y
  Ignore upper/lower case     (Y N):  N
Scan on date
  Date:     86/04/23  Compare (LT GT EQ):  __     Reset all (Y N):  N

Browse/copy member            (Y N):  Y
  Member:  APMENU1___      File:  GLCITSRC__  Library:  GLCITLIB___
Browse spooled file           (Y N):  N                            Splnbr:  *LAST
  File:     QSYSPRT____     Job:  MENUMBH___     User:  S38IMPL___  Jobnbr:  ______
Display output queue          (Y N):  N
  Outq:     QPRINT____  Library:  *LIBL_____
Screen separator line  (4 to 21):  14

Syntax checking
  When added/modified         (Y N):  Y
  From/to seqnbr:           ____.00 ____.00
  Modify source type:       CLP______
```

6. When the Edit display returns, your source member MENUxxx is at the top. APMENU1 is at the bottom. The cursor is positioned at the first line of the browse member, APMENU1.

Move the cursor on the split display to the first position of the line on the top half of the display which contains ****BEGINNING OF DATA****

Enter the line command A. The copied source code will be placed on the following line.

7. Move the cursor to the lower half of the display and enter CC in the first two positions of the sequence number 0001.00 as shown below. This line command designates the first of a series of records to be copied. Your entries should be as shown on the following display illustration.



```
EDIT     US W:1    Mbr: MENUMBH                           Scan: ____________________
FMT **   ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
A        ****BEGINNING OF DATA****
         *******END OF DATA*******



         Enter I (insert), IFff (insert under format ff), IPff (insert with
           prompt ff) or A (copy after) at cursor.  CLP ff values are:
               **,NC
         For more help, press HELP.

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
BROWSE Mbr:APMENU1    W:1   Pend: __________ Scan: ____________________
         ****BEGINNING OF DATA****
CC01.00                  PGM          /* MODULE 5  SEU EXAMPLE */
0002.00                  DCLF         FILE(APDFCK)
0003.00  START:          SNDRCVF
0004.00                  IF           COND(&OPTION = '1') THEN(CALL PGM(APCL01))
0005.00                  ELES         CMD(IF COND(&OPTION = '2') THEN(CALL +
0006.00                                 PGM(APCL02.APLIB)))
0007.00                  ELSE         CMD(IF COND(&OPTION = '3') THEN(CALL +
```

8. Press the Roll ↑ key to get to statement 0025.00, the last statement of the source code.

Note: The half of the display that is rolled is the one containing the cursor.

9. Enter CC in the first two positions of sequence number 0025.00. This designates the last record in the browse member to be copied.

10. Press the Enter key to execute the copy.

You should now see the copied records in your source member which is in the upper half of the display.

Note the code you have just added has been syntax checked as it was copied into your source member. Do not correct the statements in error. You will correct them in the next exercise when you use the Browse spooled file operation.

You could have turned off syntax checking by entering an N for the Syntax Checking operation on the Services display.

****

Next, you will use the Scan/Substitute operation to make two changes to your source code. First, remove or change library qualifications. Second, change the called program names from APCL01, APCL02, APCL03, and so on, to PRCB01, PRCB02, PRCB03, and so on.

****

11. Press CF 5 to return to the Services display.

12. Set Browse/Copy to N.

13. Set Syntax Checking to N to prevent further checking. (Lines already found in error are still highlighted.)

14. Move the cursor up to the Scan/Substitute section. Enter the scan characters .APLIB, and, because you are removing characters, enter double apostrophes ('') for the substitute characters. This tells SEU to substitute nothing once the scan characters are deleted. Press the Enter key when you have made these entries.

15. When your source member appears, move the cursor to the first position of the first record. To do this you may have to roll the displayed pages down. Scan starts at the cursor position in the page of records that is displayed.

    Press CF 7 to do a forward scan.

Because you left the default N for the "Perform all scan substitutions", the operation stops at each occurrence of .APLIB.

When the character string for which you are scanning is found, you have the choices summarized in the table shown below.

Press	Substitute	Continue Scan
Enter	yes	no
CF 1	no	no (1)
CF 5	yes	no (2)
CF 6	no	no
CF 7	yes	forward
CF 8	yes	backward

(1) Goes to the SEU Exit display.
(2) Returns to the Services display.

16. Press CF 7.

    Note the message that appears at the bottom of the display. Also note that .APLIB is removed. At each substitution occurrence, press CF 7 until the message at the bottom of the display indicates the end of the member is reached. .APLIB occurs three times in the program.

17. Press CF 5 to again get the Services display.

18. Using Scan/Substitute, change APTEST to PRLIB. This time change "Perform all scan/substitutions" to Y. Press the Enter key.

19. When your source member appears, position the cursor at the first position of the first line and press CF 7.

    Verify that APTEST is changed. It occurs three times in the program. Notice that SEU took into account the fact that PRLIB is shorter than APTEST, removed any resulting spaces, and shifted characters to left as needed. Had the substitution characters been longer, SEU would have expanded the field as necessary.

****

Now you are to change the alphabetic part of the program names from APCL to PRCB.

****

20. Press CF 5 to get the Services display.

21. In the Scan/Substitute section:

    - Enter APCL on the Scan line.
    - Enter PRCB on the Substitute line.
    - Set ''Perform all scan/substitutions'' to Y.

    Press the Enter key.

22. When your source member appears, position the cursor at the first position of the first record and press CF 7.

    Check the message at the bottom of the display to see if the substitutions were made. Ten substitutions should have been made in your source member.

23. Press CF 1 to get the SEU Exit display.

24. Select option 2 on the Exit display and, if you wish, enter a Y to print your source member. Press the Enter key.

25. When the Programmer Menu appears on the display, press CF 1 to return to the Course Menu.

26. Sign off by selecting option 90 of the Course Menu.

    Get your listing from the printer.

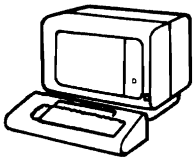
In the machine exercise you used the Browse/Copy operation to copy all of a source member into your source member. Similarly, by positioning the line commands A and CC, you can insert a portion of an existing source member anywhere in your program.

With Browse/Copy, you can even browse the original version of the member you are currently editing. For example, you may have made some changes and need to refer to the original code, or you may have accidentally deleted a statement or group of statements. Browse/Copy allows you to access the original source member and use it just like any other browse member.

You used the Scan/Substitute operation in the machine exercise to help you modify a relatively short program. You can readily see that this operation becomes more productive as the programs you need to modify become longer and more complex. If you had to read each line of code in a program to make a change, the possibility exists one entry might be missed. Scan/Substitute insures all changes are made as you specify.

Return to the Module 5 text and continue your study at the section titled "Using the Browse Spooled File Operation."

Machine Exercise 2 – Unit 2: Using the Browse Spooled File Operation



The purpose of this exercise is to familiarize you with the Browse spooled file operation of the Services display. You will see how you can use this operation to interactively correct errors found by a compile job.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise. Substitute your initials for xxx.

1. Sign on to your work station with the course password S38IMPL.

2. Select option 2 of the System/38 Implementation Topics Course Menu to display the Programmer Menu. Your personal library should still be available from the first exercise.

3. Change the **Jobd** field value on the Programmer Menu to GLCSEUJOBD.

4. Use the Create Object option (option 3) of the Programmer Menu to compile your MENUxxx program (type CLP).

5. When the Message Waiting light on your work station comes on (and the work station alarm may sound), press CF 6 to display the message.

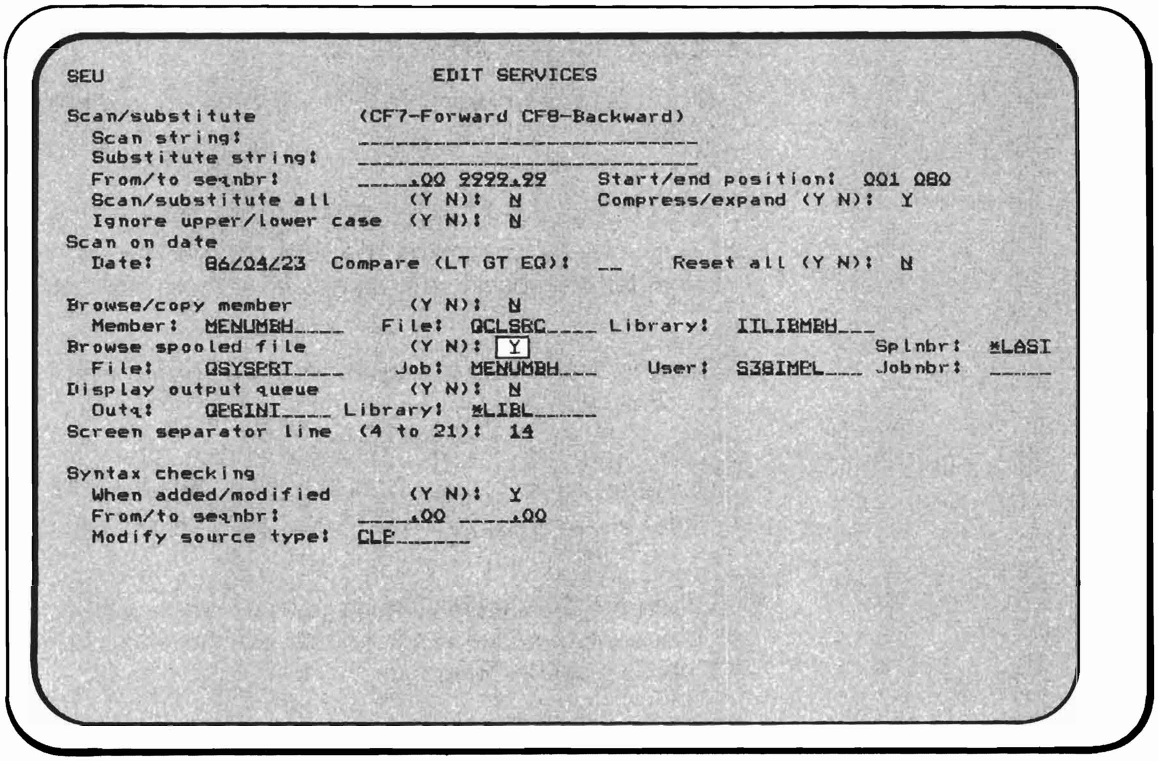
   The message should indicate your compile job ended abnormally. You did not get a successful compile.

6. Press CF 8 to return to the Programmer Menu.

7. You will now use the Browse spooled file operation to correct your program. Select option 8, Edit source, from the Programmer Menu to correct your source member, MENUxxx.

8. When the SEU Edit display with your program appears, press CF 5 to get the Services display.

9. Select the Browse spooled file operation by entering a Y in that section. An example of the completed display is shown below.

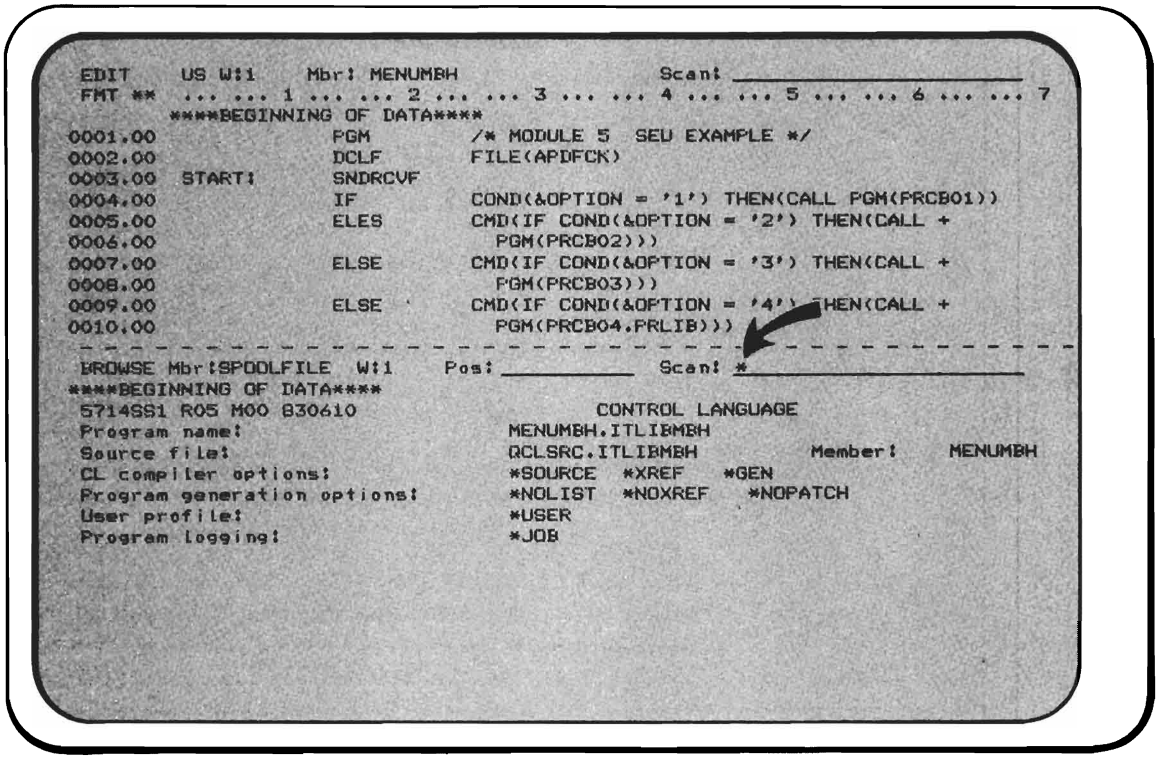


```
SEU                                  EDIT SERVICES

Scan/substitute              (CF7-Forward CF8-Backward)
  Scan string:               ____________________________
  Substitute string:         ____________________________
  From/to seqnbr:            _____.00 9999.99     Start/end position:  001 080
  Scan/substitute all        (Y N):  N            Compress/expand (Y N):  Y
  Ignore upper/lower case    (Y N):  N
Scan on date
  Date:     86/04/23  Compare (LT GT EQ):  __     Reset all (Y N):  N

Browse/copy member           (Y N):  N
  Member:  MENUMBH____   File:  QCLSRC____ Library:  IILIBMBH___
Browse spooled file          (Y N): [Y]                              Splnbr:  *LAST
  File:     QSYSPRT____    Job:  MENUMBH___    User:  S38IMPL____ Jobnbr:  ______
Display output queue         (Y N):  N
  Outq:     QPRINT____ Library:  *LIBL______
Screen separator line   (4 to 21):  14

Syntax checking
  When added/modified        (Y N):  Y
  From/to seqnbr:            _____.00 _____.00
  Modify source type:  CLP_______
```

Now press the Enter key.

10. When the split-edit display appears, the cursor is in the Pos: field of the browse member at the bottom half of the display.

    Move the cursor to the first position of the Scan: field of the browse member. Key an * in this first position, as shown below.

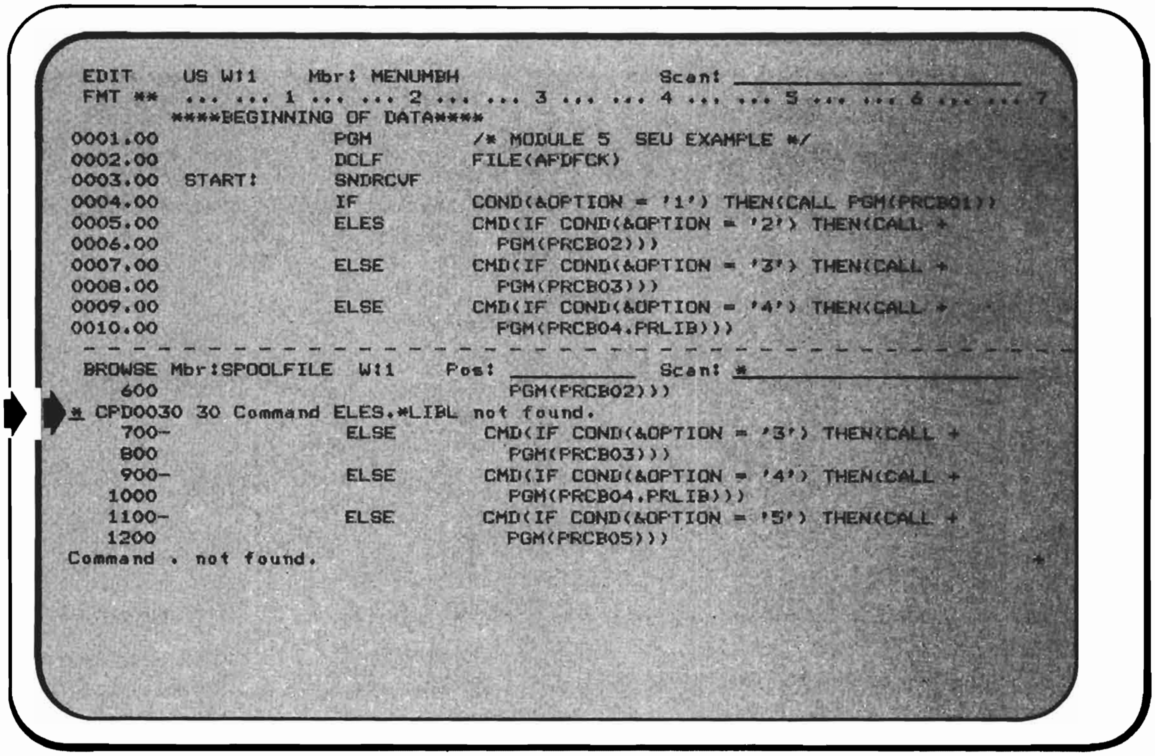


```
 EDIT     US W:1     Mbr: MENUMBH                     Scan: ______________________
 FMT **   ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
       ****BEGINNING OF DATA****
0001.00               PGM         /* MODULE 5  SEU EXAMPLE */
0002.00               DCLF        FILE(APDFCK)
0003.00  START:       SNDRCVF
0004.00               IF          COND(&OPTION = '1') THEN(CALL PGM(PRCB01))
0005.00               ELES        CMD(IF COND(&OPTION = '2') THEN(CALL +
0006.00                             PGM(PRCB02)))
0007.00               ELSE        CMD(IF COND(&OPTION = '3') THEN(CALL +
0008.00                             PGM(PRCB03)))
0009.00               ELSE        CMD(IF COND(&OPTION = '4') THEN(CALL +
0010.00                             PGM(PRCB04.PRLIB)))
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 BROWSE Mbr:SPOOLFILE  W:1     Pos: __________     Scan: *______________________
****BEGINNING OF DATA****
 5714SS1 R05 M00 830610                     CONTROL LANGUAGE
 Program name:                          MENUMBH.ITLIBMBH
 Source file:                           QCLSRC.ITLIBMBH          Member:     MENUMBH
 CL compiler options:                   *SOURCE  *XREF   *GEN
 Program generation options:            *NOLIST  *NOXREF   *NOPATCH
 User profile:                          *USER
 Program logging:                       *JOB
```

All the compile generated error messages have an * in the first position of the message record. You are asking SEU to scan the browse member – the spooled output – looking for records with an * in the first position.

11. Press CF 7 to begin the scan.

    The scan stops at the first record with an * in the first position – the first error. Your display should be similar to the one below.

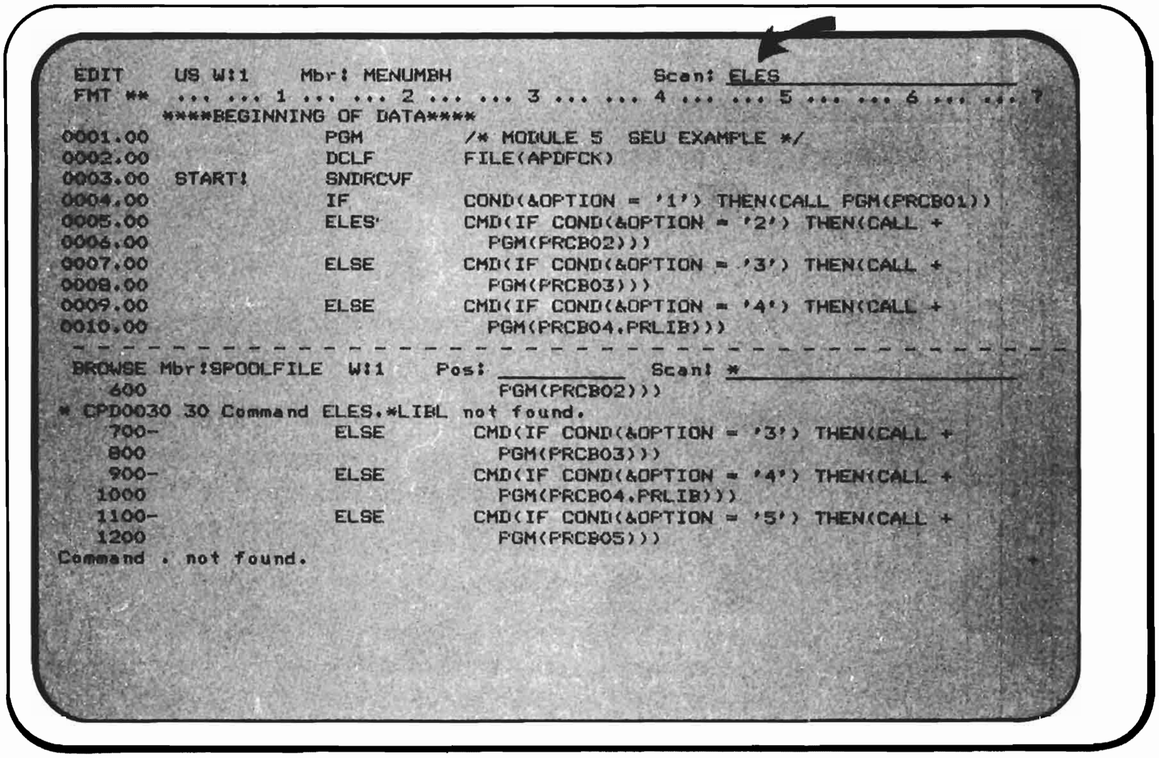


```
 EDIT    US W:1    Mbr: MENUMBH                   Scan: ____________________
 FMT **   ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
       ****BEGINNING OF DATA****
0001.00               PGM        /* MODULE 5  SEU EXAMPLE */
0002.00               DCLF       FILE(AFDFCK)
0003.00  START:       SNDRCVF
0004.00               IF         COND(&OPTION = '1') THEN(CALL PGM(PRCB01))
0005.00               ELES       CMD(IF COND(&OPTION = '2') THEN(CALL +
0006.00                            PGM(PRCB02)))
0007.00               ELSE       CMD(IF COND(&OPTION = '3') THEN(CALL +
0008.00                            PGM(PRCB03)))
0009.00               ELSE       CMD(IF COND(&OPTION = '4') THEN(CALL +
0010.00                            PGM(PRCB04.PRLIB)))
 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 BROWSE Mbr:SPOOLFILE  W:1    Pos: __________   Scan: *___________________
    600                           PGM(PRCB02)))
* CPD0030 30 Command ELES.*LIBL not found.
    700-               ELSE       CMD(IF COND(&OPTION = '3') THEN(CALL +
    800                             PGM(PRCB03)))
    900-               ELSE       CMD(IF COND(&OPTION = '4') THEN(CALL +
   1000                             PGM(PRCB04.PRLIB)))
   1100-               ELSE       CMD(IF COND(&OPTION = '5') THEN(CALL +
   1200                             PGM(PRCB05)))
Command . not found.                                                           +
```

    The cursor stops under the *. Note the error message on the display. The command ELES is not valid. It should be ELSE.

12. Move the cursor to the Scan: field of the **Edit member** at the **top** of the display.

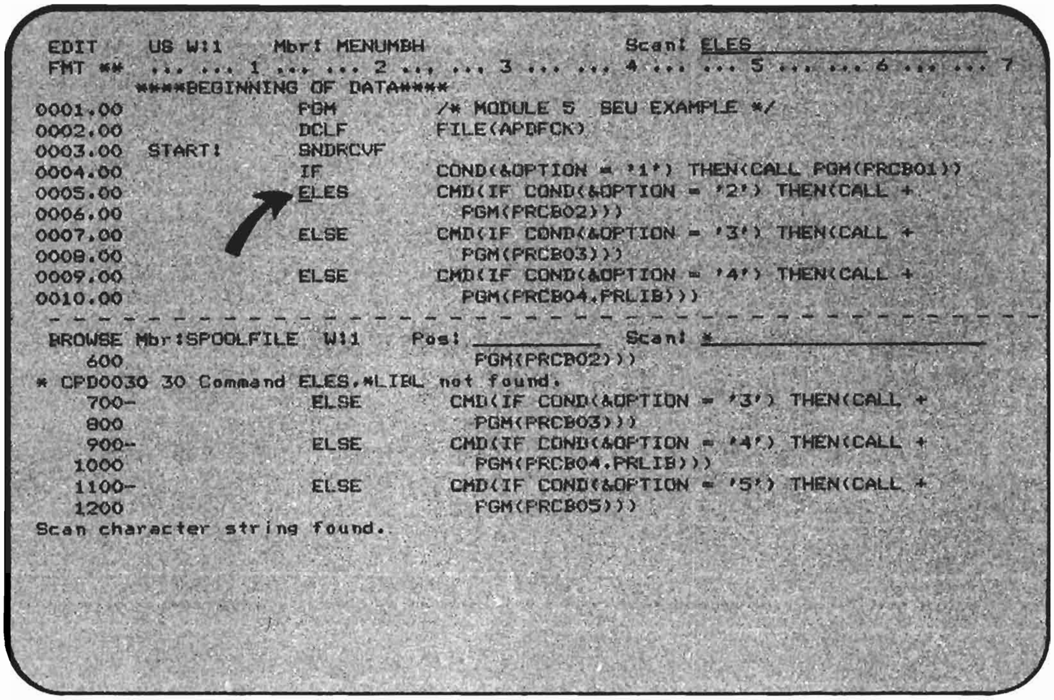
13. Key in a scan character string of ELES. You are asking SEU to find the statement in error in your program. See the example screen below.



```
 EDIT     US W:1     Mbr: MENUMBH                    Scan: ELES______________________
 FMT **  ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
        ****BEGINNING OF DATA****
0001.00               PGM          /* MODULE 5  SEU EXAMPLE */
0002.00               DCLF         FILE(APDFCK)
0003.00  START:       SNDRCVF
0004.00               IF           COND(&OPTION = '1') THEN(CALL PGM(PRCB01))
0005.00               ELES'        CMD(IF COND(&OPTION = '2') THEN(CALL +
0006.00                              PGM(PRCB02)))
0007.00               ELSE         CMD(IF COND(&OPTION = '3') THEN(CALL +
0008.00                              PGM(PRCB03)))
0009.00               ELSE         CMD(IF COND(&OPTION = '4') THEN(CALL +
0010.00                              PGM(PRCB04.PRLIB)))
 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 BROWSE Mbr:SPOOLFILE  W:1    Pos: __________    Scan: *_______________________
    600                             PGM(PRCB02)))
* CPD0030 30 Command ELES.*LIBL not found.
    700-              ELSE          CMD(IF COND(&OPTION = '3') THEN(CALL +
    800                               PGM(PRCB03)))
    900-              ELSE          CMD(IF COND(&OPTION = '4') THEN(CALL +
   1000                               PGM(PRCB04.PRLIB)))
   1100-              ELSE          CMD(IF COND(&OPTION = '5') THEN(CALL +
   1200                               PGM(PRCB05)))
Command . not found.
```

14. Press CF 7 to start the scan operation.

15. The scan locates the error in statement 0005.00 and positions the cursor at the beginning of ELES – the scan character string – as shown below. Key in the correct value – ELSE.



```
  EDIT    US W:1    Mbr: MENUMBH                 Scan: ELES
  FMT **  ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
       ****BEGINNING OF DATA****
0001.00              PGM        /* MODULE 5  SEU EXAMPLE */
0002.00              DCLF       FILE(APDFCK)
0003.00  START:      SNDRCVF
0004.00              IF         COND(&OPTION = '1') THEN(CALL PGM(PRCB01))
0005.00              ELES       CMD(IF COND(&OPTION = '2') THEN(CALL +
0006.00                           PGM(PRCB02)))
0007.00              ELSE       CMD(IF COND(&OPTION = '3') THEN(CALL +
0008.00                           PGM(PRCB03)))
0009.00              ELSE       CMD(IF COND(&OPTION = '4') THEN(CALL +
0010.00                           PGM(PRCB04.PRLIB)))
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  BROWSE Mbr:SPOOLFILE  W:1    Pos: __________   Scan: *
    600                          PGM(PRCB02)))
* CPD0030 30 Command ELES.*LIBL not found.
    700-              ELSE       CMD(IF COND(&OPTION = '3') THEN(CALL +
    800                            PGM(PRCB03)))
    900-              ELSE       CMD(IF COND(&OPTION = '4') THEN(CALL +
   1000                            PGM(PRCB04.PRLIB)))
   1100-              ELSE       CMD(IF COND(&OPTION = '5') THEN(CALL +
   1200                            PGM(PRCB05)))
Scan character string found.
```

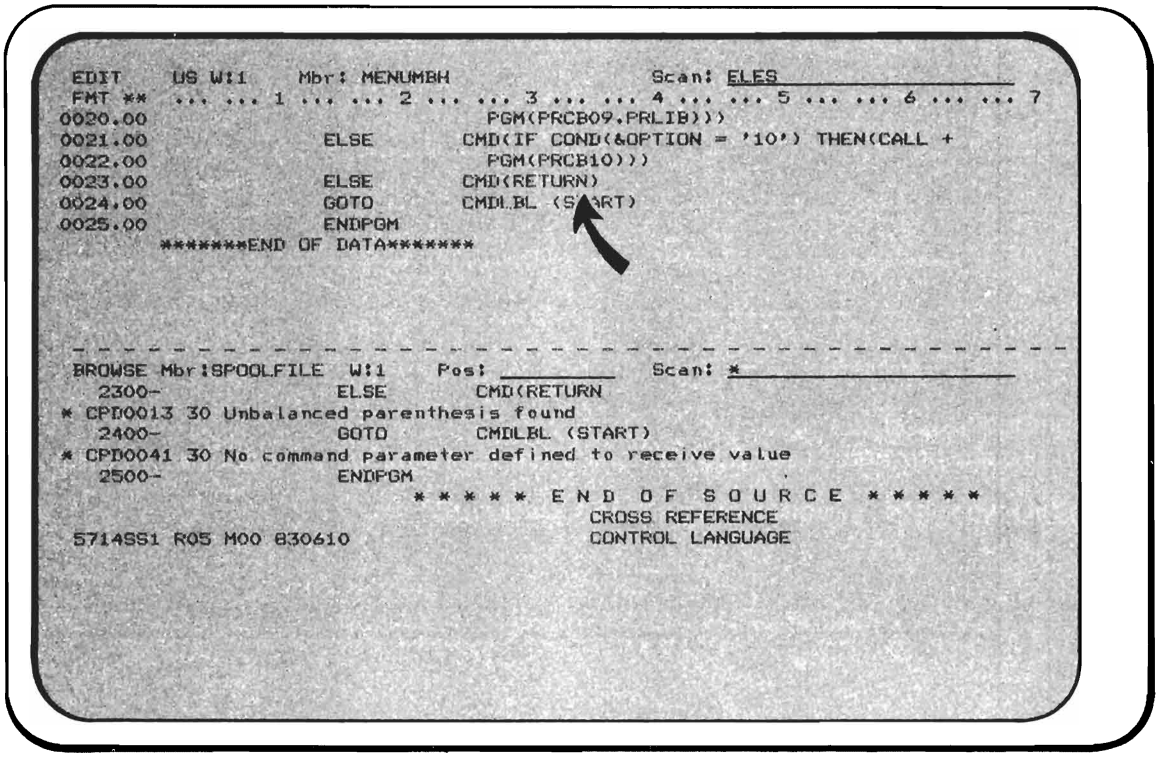
16. Move the cursor back to the Browse member and position it at the * of the error line again.

17. Press CF 7 to continue the scan for errors.

18. The next error indicates an unbalanced parenthesis. The statement in error is number 23 (2300), the one immediately ahead of the error message line.

    Move the cursor up to the Edit member section and press the Roll ↑ key until statement 23 is on the screen.

19. Position the cursor and add the missing parenthesis to correct the line, as shown below.



```
 EDIT     US W:1    Mbr: MENUMBH                    Scan: ELES
 FMT **   ... ... 1 ... ... 2 ... ... 3 ... ... 4 ... ... 5 ... ... 6 ... ... 7
0020.00                                PGM(PRCB09.PRLIB)))
0021.00              ELSE       CMD(IF COND(&OPTION = '10') THEN(CALL +
0022.00                                PGM(PRCB10)))
0023.00              ELSE       CMD(RETURN)
0024.00              GOTO       CMDLBL (START)
0025.00              ENDPGM
      *******END OF DATA*******

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 BROWSE Mbr:SPOOLFILE  W:1    Pos: _________    Scan: *
   2300-               ELSE        CMD(RETURN
* CPD0013 30 Unbalanced parenthesis found
   2400-               GOTO        CMDLBL (START)
* CPD0041 30 No command parameter defined to receive value
   2500-               ENDPGM
                              * * * * *  E N D  O F  S O U R C E  * * * * *
                                             CROSS REFERENCE
 5714SS1 R05 M00 830610                      CONTROL LANGUAGE
```

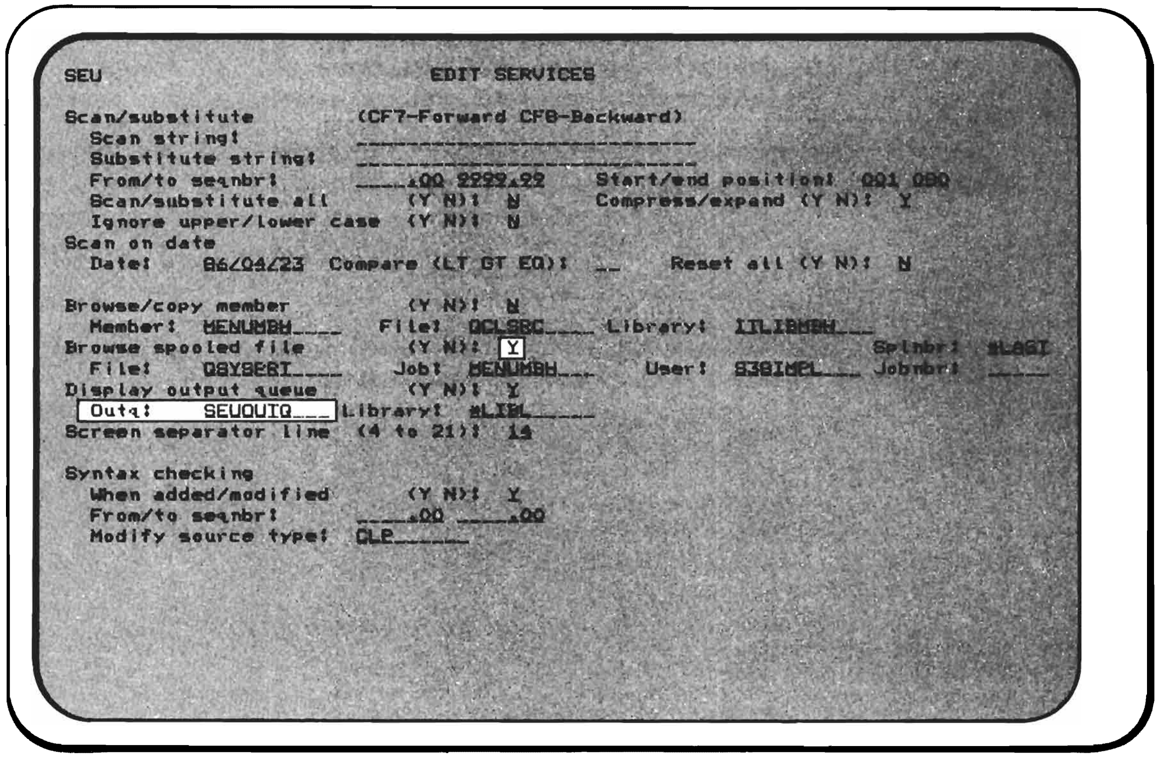
20. Looking at the Browse member, you can see the next error is at statement 24 (2400). It has a space between the CMDLBL and the parameter (START).

    Move the cursor to the invalid space on line 24 of the Edit member. Hold down the ⇧ (Shift) key and press the DEL key to delete the space.

21. Move the cursor to the **last** * error line displayed in the Browse member (the one for statement 24) and press CF 7 to see if you have any more errors.

22. When you reach the end of the Browse member, press CF 5 to get the Services display.

23. The cursor is in the Browse spooled file section of the Services display. Move the cursor to the DSPOUTQ option and Enter a Y. Type SEUOUTQ for the Outq parameter on the same line. See the display below.



```
SEU                          EDIT SERVICES

Scan/substitute            (CF7-Forward CF8-Backward)
  Scan string:             ____________________________
  Substitute string:       ____________________________
  From/to seqnbr:          ____.00 9999.99    Start/end position: 001 080
  Scan/substitute all        (Y N): N         Compress/expand (Y N): Y
  Ignore upper/lower case    (Y N): N
Scan on date
  Date:     86/04/23  Compare (LT GT EQ): __      Reset all (Y N): N

Browse/copy member           (Y N): N
  Member:  MENUMBH____   File: QCLSRC____ Library: ITLIBMBH___
Browse spooled file          (Y N): Y                          Splnbr: #LAST
  File:    QSYSPRT____      Job: MENUMBH___      User: S38IMPL___ Jobnbr: _____
Display output queue         (Y N): Y
  Outq:    SEUOUTQ___  Library: #LIBL_____
Screen separator line  (4 to 21): 14

Syntax checking
  When added/modified        (Y N): Y
  From/to seqnbr:          ____.00 ____.00
  Modify source type:      CLP______
```

Press the Enter key to get the Output Queue display.

24. On the Output Queue display, be sure the cursor is on the line with your output. Enter a 9 and press the Enter key to cancel the spooled file.

25. Press the Enter key to return to the Services display.

26. Press CF 1 to end the SEU Edit operation.

27. When the SEU Exit display appears, be sure the Option is 2 and press the Enter key.

28. Select option 3, Create object, from the Programmer Menu to compile your program, MENUxxx.

29. When the Message Light appears on your display station, press CF 6 to see the message.

30. If the message indicates your job completed normally, your program compiled successfully, press CF 8 and continue with Step 31.

    If your job did not complete normally, press CF 8 and start again with Step 7, correcting the statements you find in error.

31. Execute the command: DSPOUTQ SEUOUTQ

32. Position the cursor at the line with your output and type a 3. Press the Enter key to get the Change Spooled File Attributes display.

33. On the Change Spooled File Attributes Prompt, position the cursor at the OUTQ parameter line and type GLCOUTQ *LibL. Press the Enter key to return to the Output Queue display.

34. Position the cursor at the Option Line for your Output Queue and type a 2. Press ENTER.

    The Spooled File Attributes display shows your compile listing has been moved to the GLCOUTQ output queue in QGPL Library — an active queue — to print.

35. Press the Enter key **twice** to return to the Programmer Menu.

36. Press CF 1 to return to the Course Menu.

37. Select option 4 of the Course Menu and key your initials to remove your library.

38. Position the cursor at the option field of the Course Menu and sign off with option 90.

39. Remove your compile listing from the printer.

Your Administrator has a listing of the successfully compiled program should you desire to check your program.

Return to the Module 5 text and continue your study of the Services display.

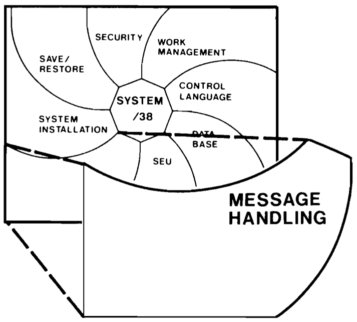
Module Summary

Source Entry Utility is a very versatile and useful tool to help you create and maintain your source members. You have used SEU in the System/38 Application Programming course and in Module 4 of this course to primarily enter and correct high level language and CL program statements. You used SEU without really dwelling on SEU. This is an indication of how easily you can use the Interactive Data Base Utility, SEU.

This module introduced some functions beyond the entry and correction of lines of code. The Services display and Exit display provide additional SEU operations you can use by simply keying options on a prompt display. In addition, Insert Skeleton provides an additional line command to make statement entry easier and faster. You can (and should) use any and all of these SEU capabilities as you continue with this course and in your future System/38 implementation efforts.

This completes your study of Source Entry Utility Additional Topics. Continue with the next module, Message Handling, by reading its Introduction on the next page.

Module 6. Message Handling



Because the System/38 is in the hands of users, you need to know how its users communicate:

- With the system, and
- With each other

You also need to be aware of how various system components communicate with each other.

System/38 provides for this communication through the use of messages. A message is a communication sent from one point in the system to another. Messages are:

- Entries you send to another system user
- Entries you send to the system itself
- Entries sent to you by other users or the system; these messages could be in response to entries you have made at your work station
- Entries sent to programs by other programs or the system

System/38, through CPF, provides the facilities for message handling. In this study module, you learn how to send and receive messages as a user and in CL programs, how to monitor for messages in a CL program, and how to create and use your own messages.

For your study of this module you will use:

- The Module 6 text
- These System/38 reference manuals:
  - System/38 Control Program Facility Programmer's Guide (SC21-7730)
  - System/38 Control Language Reference Manual (SC21-7731)
  - System/38 Operator's Guide (SC21-7735)
  - System/38 Problem Determination Guide (SC21-7876)

The next two pages in this book are provided for any notes you may wish to take as you learn about message handling on the System/38. Use them at your convenience.

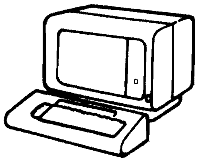
Begin your study of Message Handling in the Module 6 text now.

STUDENT NOTES: Module 6. Message Handling

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

STUDENT NOTES (continued): Message Handling

Machine Exercise – Unit 1: Basic Message Handling Commands

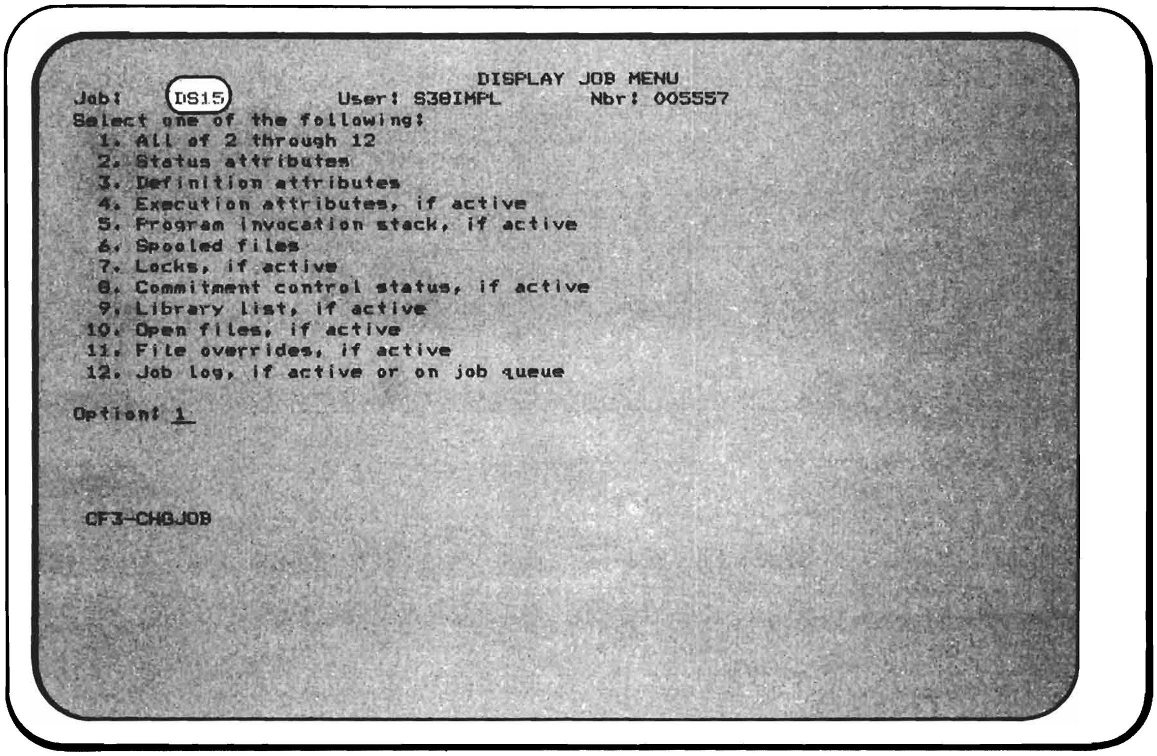


This machine exercise is intended to give you some practice using some of the message handling commands you have just learned.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

1. Sign on the display station using the password S38IMPL.
2. From the System/38 Implementation Topics Course Menu, select the option to call the Command Entry Display.
3. Determine the name of your display station's message queue. To do so, enter the CL command: DSPJOB and press the Enter key.

   A Display Job Menu, similar to the one below, appears. The work station message queue name is the same as the job name. Its location is shown on the example below. Note the name for **your** display station on the display.



```
                                   DISPLAY JOB MENU
Job:   DS15          User: S38IMPL        Nbr: 005557
Select one of the following:
  1. All of 2 through 12
  2. Status attributes
  3. Definition attributes
  4. Execution attributes, if active
  5. Program invocation stack, if active
  6. Spooled files
  7. Locks, if active
  8. Commitment control status, if active
  9. Library list, if active
 10. Open files, if active
 11. File overrides, if active
 12. Job log, if active or on job queue

Option: 1



 CF3-CHGJOB
```

Write down the name of your display station: ____________

4. When you have noted the name of your work station's message queue, press CF 1 to return to the Command Entry Display.

5. Use the SNDMSG command to send an informational message to yourself at your work station. Use any message text you wish.

6. Send a break message to yourself at your work station. Use SNDBRKMSG, make the message information only, and use any text you wish.

7. When your break message appears, press the Enter key to continue.

8. Display the messages in your work station's message queue.

9. Both your messages should be on the display. After you have displayed and reviewed the messages in the message queue, press the Enter key to continue.

10. Using the SNDMSG command, send an inquiry (*INQ) message to yourself at your work station. Use whatever message text you wish.

11. Display the messages in your work station's message queue.

12. Notice your inquiry message appears twice in the queue. The first entry is the sender's copy. The second entry is the receiver's copy, to which you must respond. Because you are both the sender and the receiver of this message, you see both copies.

    Enter a reply to the inquiry message and press the Enter key.

13. Press the Enter key to return to the Command Entry Display.

14. Change the delivery mode of your work station's message queue to break mode using the CHGMSGQ command. All your messages now automatically display.

15. Press the Enter key to return to the Command Entry Display.

16. Send an informational message to yourself at your work station. Use any message text you wish.

17. Since your work station message queue should now be in break mode, your message appears immediately. If not, repeat Steps 13 and 14. Note just the current message appears.

    Press the Enter key to return to the Command Entry Display.

18. Again display the messages in your work station's message queue.

19. All the messages you have sent to yourself during this exercise should still be in the queue and appear on the display.

    Select any one message on the display. Delete that message by positioning the cursor on the line with the message and pressing CF 6. Notice the display reappears with the deleted message gone.

20. Use CF 8 to delete all the messages in the work station message queue. The Command Entry Display appears on the screen.

21. Sign off the work station using the SIGNOFF command.

This completes the machine exercise for Unit 1. Return to your module text and continue your study with the topic "Additional Types of Message Queues."

Progress Check – Unit 1: Basic Message Handling

Read each question carefully. Record your answers in the space provided.

1. List and briefly describe four types of message queue delivery modes.

2. How can you change the delivery mode of a message queue?

3. List and briefly describe four types of messages.

4. Code the CL command to send the impromptu message 'System will be unavailable at 4:00' to all work stations. This is an important message and should interrupt any processing.

5. Which, if any, of the SNDMSG commands shown below is incorrect and why?

```
A. SNDMSG  MSG('Good Morning') TOMSGQ(WS02)
B. SNDMSG  MSG('Good Morning') TOMSGQ(*ALLWS)
C. SNDMSG  MSG('Good Morning') TOMSGQ(WS02 WS04)
           MSGTYPE(*INQ)
D. SNDMSG  MSG('Good Morning') TOMSGQ(WS02 WS04 USER1)
```

6. Assume you have signed on to a work station and the Message Waiting light is on. Code the CL command to display the message queue of that work station.

7. List three kinds of message queues.

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 2: Creating User Message Queues and Message Files

Read each question carefully. Record your answers in the space provided.

1. Code the command to create a user message queue named MYQUE. Store this queue in the library named MYLIB. Messages sent to it should be identified with the sender's job name.

2. List three characteristics of a message queue that are established when it is created.

3. For the message queue created in Question 1, code the command to change the delivery mode to default and the severity level to 30.

4. Code the command to delete the message queue created in Question 1.

5. Where are all predefined messages stored on the System/38?

6. How do you place messages in a message file?

7. List and briefly describe two control language commands you might use to maintain your own predefined messages.

____

____

____

____

____

8. What is the purpose of the CRTMSGF command?

____

____

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 3: Message Handling with CL Programs

Read each question carefully. Record your answers in the space provided.

1. List three message types, other than informational, you can specify in a CL program.

2. List the two parts of a job message queue.

3. Code the SNDPGMMSG command to send a predefined message with an identification of USR0001 to a message queue named QUExxx in a library named ITLIBxxx. The predefined message is in a message file named MSGSxxx, stored in the ITLIBxxx library. Substitute your initials for xxx.

4. Code the SNDUSRMSG command to send the message 'Do you want to update the Master File? Enter Y for Yes, N for No.' The message type should be inquiry (*INQ). Check to be sure that only Y or N are entered as a valid reply. Name the CL variable to receive the reply &RPLY.

   What additional command would have to be in the CL program in order for this SNDUSRMSG command to execute properly?

5. What command would you code to receive a message in a CL program?

   ____

   What are two items you can specify with this command?

   ____

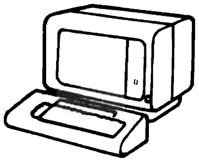
   ____

6. Code the complete CL program to accomplish the following steps.

   - When the program is initiated at a work station, execute the command you coded in Question 4.
   - Check to see if the reply is N (for No). If so, skip to the end of the program and send the predefined message USR0002. This message is in the message file MSGSxxx in the library ITLIBxxx. The message should be sent to the QUExxx message queue, also in the library ITLIBxxx. Substitute your initials for xxx. (Hint: Use the SNDPGMMSG command to send the message.)
   - If the reply is not N, execute the command you coded in Question 3.
   - Call the program named AP010, stored in the ITLIBxxx library. Substitute your initials for xxx.
   - When the program finishes, skip to the end of the program. (Hint: Be sure no message is sent, however.)

When you are finished, return to the module and review your answers with those in the text.

Machine Exercise – Unit 3: Message Handling with CL Programs



This machine exercise is intended to give you some practice using some of the message handling commands you have been studying. You will use message handling commands discussed in both Unit 2 and Unit 3.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise. Substitute your initials each time you see xxx in any of the exercise steps.

1. Sign on to the work station using the password S38IMPL.

2. Select the option from the System/38 Implementation Topics Course Menu to create a library. Do not forget to enter your initials.

   Note: As a result of choosing the create library option, the Programmer Menu appears on the screen. You may use this menu or switch to the Command Entry Display to enter the commands for this exercise. The choice is yours. (Since you are primarily entering commands, the Command Entry Display would probably be easier to use.) Remember, to get to the Command Entry Display from the Programmer Menu, use CF 3. To get back to the Programmer menu, use CF 1.

3. Create a user message queue using CRTMSGQ. The name of the queue should be QUExxx in the library ITLIBxxx. Identify message senders by their job name.

4. Create a user message file named MSGSxxx in the library ITLIBxxx with CRTMSGF.

5. Add two messages to the MSGSxxx message file in ITLIBxxx using the ADDMSGD command. Use the following for message id and text:

MSGID	MSG
USR0001	'Master file update beginning'
USR0002	'Master file update skipped'

6. Enter the CL program from Question 6 of the Progress Check. Name the program CLPxxx. (Hint: Use Option 8 of the Programmer Menu with CLPxxx as the first parameter and CLP as the Type entry.)

7. Create an object for the CL program you just entered. Use Option 3 of the Programmer Menu to do this. Be sure the first parameter is CLPxxx and the Type entry is CLP.

8. When the Message Waiting light appears on your work station, press CF 6 to display the message.

   - If your job completed normally, continue with Step 9.
   - If your job did not complete normally, use your program listing to determine the errors and start again with Step 6.

   Press CF 8.

9. Set your user message queue, QUExxx, to notify mode, using the CHGMSGQ command.

10. Execute your program, CLPxxx. Respond to the message by indicating Yes, you do want to update the master file. This message display is from the **program's** external message queue. Notice the heading of the display – Program Messages.

    Note: If your program does not successfully execute for any reason (for example, you do not get the message asking if you want to update), determine your error and begin again with Step 6.

11. When the message ''AP010 has executed'' appears on the display, press the Enter key to return to the Command Entry Display or Programmer Menu, whichever you are using. Again, this message has come from the program's external message queue to the screen.

12. Run your program, CLPxxx, again. This time, respond to the message by indicating No, you do not want to update the master file.

13. Run your program again, responding to the message with an invalid reply (something other than Y or N). Note what happens.

    Enter a valid response to continue.

14. If you entered a Y for Step 13 to continue, press the Enter key when the message ''AP010 has executed'' appears on the display to get the Command Entry Display or Programmer Menu.

    If you entered an N for Step 13 to continue, the Command Entry Display or Programmer Menu appears automatically.

15. Display your user message queue, QUExxx. Note the messages it contains.

16. Remove all the messages from your user message queue with CF 8.

17. Sign off the display station, If you have not already done so, collect your listing from the printer.

> When you have finished the exercise, you may review any part of the unit before continuing. When you are ready, return to the module text and begin the next unit — ''Message Handling.''

Progress Check – Unit 4: Problem Determination and Message Handling

Read each question carefully. Record your answers in the space provided.

1. List the three types of messages you can monitor for in a CL program.

2. List and briefly describe the two levels of MONMSG commands you can use to monitor for messages.

3. Assume the two CL commands below appear together in a CL program. Explain briefly what they are doing.

```
CHKOBJ   OBJ(APUPD.APLIB) OBJTYPE(*PGM)
MONMSG   MSGID(CPF9801) EXEC(GOTO NOPROG)
```

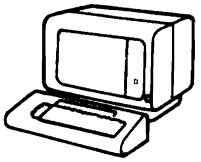
4. List two System/38-generated logs you can use to aid you in problem determination.

5. Code the CL commands to accomplish the following: (Note: You are **not** coding a complete CL program, just some commands. You should, however, be able to incorporate them into a valid CL program.)

   - Declare two decimal variables, &P1 and &P2. Variable &P1 is passed via a CALL statement and is to be defined with a length of 15 positions and 5 decimal places (the default values). &P2 should be defined as 3 positions. Place any initial numeric value you wish in &P2.
   - Change the variable &P1 by dividing &P2 by &P1.
   - Monitor the CHGVAR command for a zero divide escape message, MCH1211, using MONMSG.
   - If MCH1211 is received, change variable &P1 to one (1) and re-execute the divide operation.

When you are finished, return to the module and review your answers with those in the text.

Machine Exercise – Unit 4: Problem Determination with Message Handling



This machine exercise is intended to give you some additional practice with message handling commands. You will use message handling commands discussed throughout this module as well as other CL commands you have seen.

Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. If you wish, make any notes you feel will aid you as you perform the exercise. Be sure to substitute your initials for xxx.

1. Sign on to the work station using the password S38IMPL.

2. Select the option from the System/38 Implementation Topics Course Menu to display the Programmer's Menu. Be sure to enter your initials. (Note: Your own personal library should still be available from the previous exercise. If not, select the option from the Course Menu to create your library.)

3. Modify your CL program, CLPxxx, from Step 6 of the previous exercise (Question 6 from the Progress Check of Unit 3) to do the following:

   After AP010 is done:

   Ⓐ Use the CHKOBJ command to see if the program (*PGM) AP011 is in your library (ITLIBxxx). Monitor for the message CPF9801 (Object not found) coming from CHKOBJ.

   - If CPF9801 is received, send the message 'AP011 not in the library' to your message queue (QUExxx) and copy AP011 into your library from the GLCITLIB library using the command (Create Duplicate Object):

```
CRTDUPOBJ  OBJ(AP011) FROMLIB(GLCITLIB) +
           OBJTYPE(*PGM) TOLIB(ITLIBxxx)
```

   (Hint: Use EXEC(DO) on the MONMSG command to control the SNDMSG and CRTDUPOBJ commands)

   Ⓑ Call the program AP011, located in your library.

   Use option 8 of the Programmer Menu with the correct parameter and Type entry to make your program changes.

   The coded solution for Unit 3 Progress Check Question 6 is shown on the next page for your reference.

```
         PGM  /* Progress Check Question 6 */
         DCL  VAR(&RPLY) TYPE(*CHAR) LEN(1)
         SNDUSRMSG  MSG('Do you want to update the Master +
                        File? Enter Y for Yes, N for No.') +
                    MSGTYPE(*INQ) +
                    VALUES(Y N) +
                    MSGRPY(&RPLY)
         IF  COND(&RPLY *EQ N) THEN(GOTO SKIP)
         SNDPGMMSG  MSGID(USR0001) +
                    MSGF(MSGSxxx.ITLIBxxx) +
                    TOMSGQ(QUExxx.ITLIBxxx)
         CALL  AP010.ITLIBxxx
         GOTO  END
SKIP:    SNDPGMMSG  MSGID(USR0002) +
                    MSGF(MSGSxxx.ITLIBxxx) +
                    TOMSGQ(QUExxx.ITLIBxxx)
END:     ENDPGM
```

4. Create an object for your CL program you just modified. Use option 3 of the Programmer's Menu to do this. Be sure to use the correct parameter and Type entries.

   If you are informed an object with the same name already exists, press the Reset key and CF 11 to continue creating the object. The old version is replaced with your modified version.

5. When the Message Waiting light appears on your work station, press CF 6 to display the message.

   - If your job completed normally, press CF 8 and continue with Step 6.
   - If your job did not complete normally, press CF 8. Use your program listing to determine the error and start again with Step 3.

6. Execute your program, CLPxxx, located in your library.

7. When you are prompted, select the choice that indicates yes, you do want to update the Master File.

8. When the message 'AP010 has executed' appears on the screen, press the Enter key to continue processing.

9. Display the System Request Menu. To do so:

   Ⓐ Hold down the Shift key and press the Sys Req/Attn key

   Ⓑ Press the Enter key

   The System Request Menu should now be on the screen.

10. Display the messages in **your** message queue. To do so:

    Ⓐ Key a 4 for the Option,

    Ⓑ Move the cursor to the Parameters field and key the qualified name of your message queue in your library (QUExxx.ITLIBxxx),

    Ⓒ Press the Enter key

    Was AP011 in your library? The message display should tell you if it was not.

11. Press the Enter key to end the message display and return to the System Request Menu.

12. Press CF 1 to return to your interrupted program. The Program Message display should be on the screen.

13. When the message 'AP011 has executed' appears on the screen, press the Enter key. The Programmer Menu should appear on the screen.

14. Display your message queue again.

15. Delete all the messages from your message queue with CF 8.

16. Return to the Course Menu by pressing CF 1.

17. Select the option from the Course Menu to remove your exercise materials.

18. Sign off the display station. Be sure to get your listings from the printer.

The Administrator has a possible solution for the modified CL program should you want a copy.

This completes the machine exercise. Continue with the Module Summary for Message Handling on the next page.

Module Summary – Message Handling

The System/38 provides for message handling in many ways. Message handling is used to send and receive messages and for system/user communication. This module introduced you to System/38 message handling concepts, terms, functions, and commands.

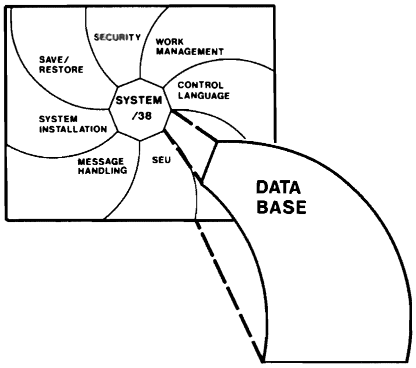
Specifically, this module made you aware of

- Different kinds of message queues
  - Work station
  - User
  - Job
- Message queue delivery modes
  - Notify
  - Break
  - Hold
  - Default
- Various types of messages, including
  - Informational
  - Inquiry
  - Completion
  - Request
  - Break
  - Escape
- How you can send and display messages
  - SNDMSG
  - SNDBRKMSG
  - DSPMSG
- How you can create, change, and delete message queues
  - CRTMSGQ
  - CHGMSGQ
  - DLTMSGQ
- How you can create, change, and delete your own user-defined messages
  - CRTMSGF
  - ADDMSGD
  - CHGMSGD
  - DSPMSGF
  - DSPMSGD
  - RMVMSGD
- How you can send and receive messages in a CL program
  - SNDUSRMSG
  - SNDPGMMSG
  - RCVMSG
- Message handling provisions you can use for problem determination, including
  - Job and history logs
  - MONMSG for escape, notify, and status messages
  - System Reply List

Message handling functions and capabilities should be an integral part of any application you implement on your System/38.

This completes your study of message handling. When you are ready, continue with the next module, Data Base Topics, by reading its Introduction on the next page.

Module 7. Data Base Topics



In the System/38 Application Programming course, you learned about physical and logical files, and how to use data description specifications when creating data base files.

In this module, you will learn additional functions available in data management. You will learn about ways to improve the performance of file operations and ways of locking files from other running jobs. You will learn about the many powerful capabilities of the copy command over and above just copying a file. You will see how you can prevent the loss of data when a file is disabled. You will learn how you can ensure that every file operation is successful in a multi-file transaction before the file changes are made permanent.

You will use the following materials in studying Data Base Topics:

- The Module 7 text
- The reference manuals listed below:
  1. IBM System/38 Control Language Reference Manual (SC21-7731)
  2. IBM System/38 Control Program Facility Programmer's Guide (SC21-7730)

Begin your study of Data Base Topics in the Module 7 text now.

STUDENT NOTES: Module 7. Data Base Topics

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

STUDENT NOTES (continued): Data Base Topics

Progress Check – Unit 1: File Sharing

Read each question carefully. Record your answers in the space provided.

For Questions 1 through 5, indicate whether each statement is true (T) or false (F)

_____ 1. If one job allocates a file member *EXCL, another job can concurrently access the file member with an allocation of *SHRUP.

_____ 2. If one job allocates a data area with *SHRNUP, another job can concurrently read the data area with an allocation of *SHRRD.

_____ 3. Any outstanding lock states are removed at the end of a routing step.

_____ 4. If a file member is allocated *SHRRD, another job can concurrently read a record of the file member without issuing an allocation command.

_____ 5. If an object is allocated more than once in a routing step you can remove the lock states with one deallocate command.

6. List two benefits of sharing an open data path.

____

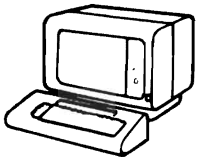
____

7. In addition to an open data path, what other file resource can be shared?

____

When you are finished, return to the module and review your answers with those in the text.

Machine Exercise – Unit 2: Using the Copy Commands



Read through the entire exercise before you go to the work station so you know in advance what you are expected to do. Make any notes, if you wish, to aid you when you are doing the exercise.

This exercise demonstrates some of the more important capabilities of the CPYF command. Plus, you will use the CPYSRCF command.

In this exercise, you create a new vendor master file after modifying its description (source member). You then move the data from the existing file to the new file. The primary purpose of this part of the exercise is to illustrate the use of the FMTOPT parameter.

Next, you print the records in the new file.

Finally, you print selected records from the file.

Your Administrator has a solution for the copy commands you enter in doing the exercise. You should make every effort to do the exercise before looking at the solution.

In doing the exercise, you should consider using CF 4 to get the command prompting displays for entering the parameter values. Always check your entries before pressing the Enter key.

Note: Substitute your initials for xxx in the instructions.

1. Sign on to a System/38 work station using the course password S38IMPL.

2. When the System/38 Implementation Course Menu appears, select option 1 and enter your initials to create your library, ITLIBxxx.

3. When the Programmer Menu appears on the screen, get the Command Entry Display by pressing CF 3.

4. Using the CPYSRCF command, copy the member named APPVEND from the source file GLCITSRC in the library GLCITLIB to your source file QDDSSRC in your library, ITLIBxxx. This is the DDS source for the accounts payable vendor master file, which is a physical file.

   24 records should be copied.

5. Return to the Programmer Menu with CF 1.

6. On the Programmer Menu, select Edit source with option 8 and enter the name APPVEND for the first parameter (Parm:) and PF for the Type.

7. When the source for APPVEND is displayed, do the following:
   a. Move the cursor down to the field named VNDZIP. Use the line code P to get the prompt for this specification. Enter 9 in the length (Len) column.
   b. Delete the specification for the field named VNDCLS.
   c. Add a field after the last field. Name the field TRNDT and give it a length of 6 with 0 decimal places. Leave the data type column blank to default to packed.

8. Exit source editing by pressing CF1.

9. When the SEU EXIT display appears, update the member with option 2. Also, you may want to print the member.

10. From the Programmer Menu, select option 3 to create a new physical vendor master file (APPVEND) in your library. Get the listing from the printer and verify that the compile was successful.

11. Press CF 3 to get the Command Entry Display.

12. Copy the records from the APPVEND file in the GLCITLIB library to the new and modified APPVEND file in your library, ITLIBxxx. The two key parameters you will use in this exercise are FMTOPT and MBROPT. You may want to review them in Unit 2 of the Module 7 text or in the Control Language reference manual (SC21-7731).

    A total of 47 records should be copied from the file.

13. Use the CPYF command to list the first six (6) data records in the new file, APPVEND, in your library, ITLIBxxx. Specify PRTFMT (*HEX). Get and examine your listing from the printer.

14. Enter a CPYF command to list records from APPVEND in ITLIBxxx with a service rating (the SRVRTG field) of A and a delivery rating (the DELRTG field) of A. Specify PRTFMT(*CHAR). Get and examine your listing from the printer. The listing should have 15 records.

15. Press CF 1 to return to the Programmer Menu.

16. Press CF 6 to display messages.

17. Press CF 8 to return to the Programmer Menu.

18. Press CF 1 to return to the Course Menu.

19. Select option 4 and enter your initials on the Course Menu to remove the exercise material.

20. Sign off the display. Be sure you have removed all your listings from the printer.

When you are finished with the exercise, continue with Unit 3, Using Journaling, in the Module 7 text.

Progress Check – Unit 3: Using Journaling

Read each question carefully. Record your answers in the space provided.

Indicate whether each of the following statements is true (T) or false (F).

_____ 1. A journal entry is made for every file operation.

_____ 2. A journal entry is made for a record before it is written to the data file.

_____ 3. Journaling effectively gives a force write ratio of 1.

_____ 4. Journaling prevents a file from being damaged.

_____ 5. Specifying before-images is valid only with file update.

_____ 6. You can journal a file to more than one journal.

_____ 7. You can only specify physical files to be journaled.

_____ 8. An entry is made in a file's journal when it is saved.

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 4: Using Commitment Control

Read each question carefully. Record your answers in the space provided.

Indicate whether each of the following statements is true (T) or false (F).

_____ 1. Journaling is a prerequisite for commitment control.

_____ 2. Program logic is used to determine if a multi-file transaction should be committed.

_____ 3. The files whose operations are to be committed are locked from use by other jobs when the Begin Commitment Control (BGNCMTCTL) command is executed.

_____ 4. You have to specify journal before-images in order to use commitment control.

When you are finished, return to the module and review your answers with those in the text.

Module 7 Summary – Data Base Topics

In this module you learned about additional data base capabilities available in the Control Program Facilities.

With file sharing you added to your knowledge of sharing file resources by studying lock states and the sharing of an open data path. You learned that sharing an open data path, which is very simple to implement, results in the saving of memory and better performance during program execution.

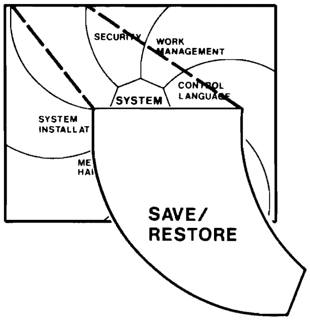
In your in-depth study of the Copy File (CPYF) command you learned about its many capabilities, probably the most important of which is to copy data to a file with a format different from the original file. Also, you learned that the CPYF command has very powerful record selection capabilities.

You learned that you can use journaling to minimize the risk of losing data if a file were damaged. You saw how journaling worked and learned how to code the commands to implement and manage journaling.

You learned that commitment control may be used for a transaction that involves multi-file operations and gives you the capability of having CPF restore the files to their original condition if the transaction cannot be completed successfully.

This completes your study of Data Base Topics. Continue the course by reading the Module 8 Introduction on the next page.

Module 8. Save/Restore



Included with CPF are the functions you need to save objects outside your system and later restore them to your system. CPF provides functions to:

- Save objects from the system by writing a copy of the objects to storage outside the system
- Restore objects to the system that have previously been saved

The **Save/Restore** operations provide you with a means to transfer information between internal (auxiliary) storage and external diskettes or magnetic tape. These save/restore facilities should be used to establish procedures to use to back up your system. These procedures should be designed and implemented as an integral part of your system. The use of save/restore functions is essential for the proper maintenance of your system's files and objects.

This module introduces you to the System/38 Save/Restore facilities.

For your study of this module you will use:

- The Module 8 text
- System/38 Control Program Facility Programmer's Guide (SC21-7730)

The next pages in this book are provided for any notes you may wish to take as you learn about save/restore on the System/38. Use them at your convenience.

Begin your study of Save/Restore in the Module 8 text now.

STUDENT NOTES: Module 8. Save/Restore

As you proceed through the study module, use these pages to record any notes you feel will help you understand the topic.

STUDENT NOTES (continued): Save/Restore

Progress Check – Unit 1: Save/Restore Concepts

Read each question carefully. Record your answers in the space provided.

1. List any four of the six items you can save and restore with the Save/Restore functions of the System/38.

2. Code the CL command and its parameters needed to initialize the diskettes in magazine slot *M2 to save/restore format. The diskettes should have a volume identifier of SYSTM. Check to see if the diskettes contain any active files.

3. As a result of executing the initialization command of Question 2, what is the complete volume identifier of the diskette in slot 5 of the magazine?

4. Code the CL command to list the names of all the files on the diskettes in the first two single slots of the diskette drive.

Skip Questions 5 and 6 if you did not study the section on Magnetic Tape in the unit.

5. Code the CL command and its parameters needed to initialize the tape mounted on the tape device named QTAPE3. Give the tape a volume label of SYSTM and check to see if any active files are on the tape.

6. Code the CL command to display the names of any files recorded on the tape mounted on the tape device QTAPE3.

____

7. List 3 pieces of save/restore status information the System/38 keeps about an object.

____

____

____

____

8. What happens to an object in auxiliary storage when you save it to diskette or tape?

____

____

____

What happens to the version of an object in auxiliary storage when you restore a saved copy of it?

____

____

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 2: Saving Information

Read each question carefully. Record your answers in the space provided. For any command coding, code only the parameters you feel are necessary to answer the question **as it is stated** and use parameter defaults as appropriate.

1. List the four CL commands used to save objects and libraries.

   ____

   ____

   ____

   ____

2. List three objects that cannot be saved.

   ____

   ____

   ____

3. Code the command to save all the objects in the library GLLIB that have changed since the last time the library was saved. The objects should be saved on diskettes named GLSAV.

   ____

   ____

4. Code the command to save the libraries GLLIB, ARLIB, and INLIB on diskettes. Use whatever diskettes are in the diskette drive, but clear any uncleared diskettes found. Check to see if all the objects can be allocated to the save operation.

   ____

   ____

5. Code the command to save the programs GLLIST, GLUPD, and GLCHK in the library GLLIB on diskette. Free the storage of these objects when the save is complete. Use whatever diskettes are in the diskette drive and clear any uncleared diskettes.

____

____

6. List the steps you would perform to save your entire system.

____

____

____

____

____

____

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 3: Restoring Information

Read each question carefully. Record your answers in the space provided. For any command coding, code only the parameters you feel are necessary to answer the question **as it is stated** and use parameter defaults as appropriate.

1. What two restore commands are used only when you are restoring your saved system?

   ____

   ____

2. Code the command to restore the saved library ARLIB from diskette to a library named ARAPP. Restore the version of ARLIB saved on 05/01/84 to a diskette with a volume identifier of ARSAV.

   ____

3. Code the command to restore all the programs beginning with AP and GL. They were originally saved from and are to be restored to a library named FINANCE. Restore only those objects whose storage was freed when they were saved. The objects were saved on diskette.

   ____

4. Code the command to restore user profiles from the tape mounted on the tape device named QTAPE2.

   ____

5. List in the correct order the steps you would perform to restore your entire system.

   ____

   ____

   ____

   ____

   ____

   ____

When you are finished, return to the module and review your answers with those in the text.

Progress Check – Unit 4: Save/Restore Considerations

Read each question carefully. Record your answers in the space provided.

1. List four factors that influence overall performance with save/restore functions.

   ____

   ____

   ____

   ____

   ____

2. List two save/restore functions that require the System/38 to be dedicated.

   ____

   ____

   ____

3. List four factors you should take into consideration when you develop your save/restore strategy.

   ____

   ____

   ____

   ____

   ____

4. What happens if you restore an object whose owner does not exist in the system?

   ____

   What happens if you attempt to restore an object over an existing object and the owner name in the system and the owner name recorded with the saved version are different?

   ____

When you are finished, return to the module and review your answers with those in the text.

Module Summary – Save/Restore

The System/38 save/restore functions provide you with the capability to copy vital system information onto diskette or tape. Should CPF programming, user programs, or data be lost as the result of a failure, you have the procedures to restore them to the system in order to continue your operation.

To provide a backup of your system, you use the Save commands (SAVLIB, SAVOBJ, SAVCHGOBJ, SAVSYS) to copy information to diskettes or magnetic tape. To return the saved information to the system, you use the Restore commands (RSTLIB, RSTOBJ, RSTUSRPRF, RSTAUT). Remember, you must re-install CPF to restore your entire system, however.

Save/restore functions use either diskette or tape. Diskettes must be initialized to the save/restore format using the INZDKT command. Tapes must be initialized with a standard label using INZTAP. If you are saving your entire system to tape, remember to have one diskette available to receive CPF programs to start the restore operation.

You must design a save/restore strategy for your System/38. As you do so, you must evaluate the save/restore options, online backup and journaling. You also must consider performance and security factors. No single approach addresses all your application needs. Study your operational requirements and resources and ask yourself "How long can I afford to be down?"

When you are ready, continue with the Course Summary on the next page.

Implementation Topics Summary

If you look back over the modules you studied in this course, you can see the role each topic plays in the implementation of your System/38:

1. System Installation...device configuration, program product installation, and system tailoring to get a usable system
2. Security...how to control access to your system and protect it from unauthorized use
3. Work Management...how work management operates and how you can change various work management objects
4. Control Language Programming...how to code CL programs and use them to control an application
5. Source Entry Utility...how to use additional SEU functions, including Services operations and Exit display options
6. Message Handling...communication between users, the system and a user, and a program and a user
7. Data Base...file creation and usage considerations
8. Save/Restore...protecting your programs and data once your system and applications are implemented

You are now aware of the tasks necessary to implement your System/38. You are familiar with the available reference material to help you complete these tasks and to provide further study. You should now be ready to select your operating environment and begin to implement it.

You have no more Progress Checks. Your next 'test' is the actual implementation of your System/38. This course was intended to help you identify and plan for the pieces necessary for the implementation effort. With the knowledge you now have, your System/38 reference manuals as guides, and your IBM representative as a consultant, you should have a successful installation.

> At this time, complete any notes here in your Student Materials Book, return any modules to your Administrator, and complete any end-of-course activity.

Congratulations on the completion of this important part of your System/38 education. Refer to the System/38 curriculum flowchart in Section II for your next course.

Appendix A

System/38 Curriculum Overview

Guided Learning Center Course Offerings

The following list shows the major topics of each module for the available System/38 Guided Learning Center courses.

System/38 Fundamentals

Module 1. Course Introduction
- Hardware Overview
- Software Overview

Module 2. Data Base Data Management
- Data Base Data Management Introduction
- Building the Data Base
- Additional Topics on Building Data Base

Module 3. Work Station Data Management
- Work Station Overview
- Work Station Implementation

Module 4. Control Language
- Control Language and CL Programs

Module 5. Programmer Services – Utilities
- Interactive Data Base Utilities

Module 6. Programmer Services – Languages and Aids
- High Level Languages
- Programmer Productivity Aids

Module 7. Object Management and Work Management
- Object Management
- Work Management

Module 8. System Service Facilities
- System Service Facilities (Security, Save/Restore)
- Conversion and Installation Facilities

System/38 Application Programming

Module 1. Course Introduction
- Course Overview
- IBM System/38 Reference Library
- Familiarization Exercise

Module 2. System/38 Review
- System/38 Review – CPF
- Review Questions

Module 3. System/38 Operation
- The Command Entry Display
- The Programmer Menu

Module 4. Building the Data Base
- Review of Data Base Concepts
- Data Description Specifications
- Transferring DDS to Machine Readable Media

Module 5. Introduction to Interactive Processing
- Fundamentals of Interactive Processing

Module 6. Data File Utility
- Introducing DFU
- How to Design/Create a DFU Application
- How to Execute a DFU Application
- How to Change a DFU Application
- How to Delete a DFU Application

Module 7. Display Files
- Concepts of Display Files
- Creating Simple Display Files
- Creating More Complex Display Files

Module 8. RPG III Interactive Programs
- RPG III for Simple Display Files
- RPG III for More Complex Display Files

Module 9. COBOL Interactive Programs
- COBOL for Simple Display Files
- COBOL for More Complex Display Files

Module 10. Introduction to Batch Processing
- Fundamentals of Batch Processing
- How to Run a Batch Job

Module 11. RPG III Batch Programming
- RPG III for Batch Programming

Module 12. COBOL Batch Programming
- COBOL for Batch Programming

System/38 Application Programming

Module 13. The Query Facility
- Introducing Query
- Application Creation and Execution
- Modifying a Query Application

Module 14. DEBUG Facilities
- Purpose of Debug Mode
- Operations in Debug Mode

Module 15. Control Language Programs
- Concepts of Control Language Programs
- CL Programs for Menu Driven Applications

Module 16. Screen Design Aid
- Introducing SDA
- SDA for Testing Display Files
- SDA for Creating Menus
- SDA for Creating Display Formats

System/38 RPG III and Structured Programming

Module 1. Course Introduction
- Course Overview
- Prerequisite Review

Module 2. Procedural File Operations
- File Processing
- File Processing Operation Codes

Module 3. Structured Programming Concepts
- Fundamentals of Structured Programming

Module 4. Structured Programming Implementation
- Structured Programming Operation Codes
- Developing a Structured Program

Module 5. Program Compilation and Testing
- Compiling, Testing, and Debugging Programs

Module 6. Subprograms
- RPG III External Subroutines

Module 7. Subfile Programming
- Subfile Coding – Inquiry
- Subfile Coding – Update

Module 8. Data Areas
- Use and Coding of Data Areas

Module 9. Data Structures
- Use and Coding of Data Structures

Module 10. Handling Exception Errors
- How to Handle Exception Errors in Programs

Module 11. Additional RPG III Functions
- RPG Enhancements
- Multiple Device Files
- Commitment Control

Module 12. Program Described Work Station Files
- How to Describe Work Station Files Within a Program

Module 13. Printer Device Files
- Externally Described Printer Files

System/38 Implementation Topics

Module 1. System Installation
- Pre-installation
- Installing Your System
- Post-installation

Module 2. Security
- Physical Security
- System Security
- System Security Implementation

Module 3. Work Management
- Concepts
- Interactive Job Processing
- Batch Job Processing
- Considerations

Module 4. Control Language Programming
- Using and Coding CL Programs
- Communicating Between Programs
- Creating Objects with CL Programs
- Character String Operations
- Data Areas
- Retrieving System Values and Job Status

Module 5. Source Entry Utility Additional Topics
- Services Display Operations
- Exit Display Options
- Insert Skeleton

Module 6. Message Handling
- Basic Message Handling
- Creating User Message Queues and Message Files
- Message Handling with CL Programs
- Problem Determination with Message Handling

Module 7. Data Base Topics
- File Sharing
- Copy Commands
- Journaling
- Commitment Control

Module 8. Save/Restore
- Save/Restore Concepts
- Saving Information
- Restoring Information
- Save/Restore Considerations

Appendix B. System Configuration Objects

To verify the names of the system configuration objects for your system configuration (*once CPF is loaded and started*), use the Display Device Configuration (DSPDEVCFG) command.

The device descriptions for the system console and the diskette magazine drive are supplied with CPF.

For additional information:

- Use the Display Device Description (DSPDEVD) command to display the parameter values specified for a particular device description.
- Use the Display Control Unit Description (DSPCUD) command to display the parameter values specified for a particular control unit description.

The following tables show the commands used to create the system configuration objects. For example, to create a diskette magazine drive, enter:

```
CRTDEVD  DEVD(QDKT)  DEVADR(000012)
   DEVTYPE(72MD)  MODEL(1001)  TEXT('Diskette Drive')
```

CONTROL UNIT DESCRIPTIONS

The following chart omits parameters for which the default is taken.

CRTCUD Command Parameters	**Work Station Controllers**				**Magnetic Tape Control Units**	
CUD	QWSC1	QWSC2	QWSC3	QWSC4	QTAPE	QTAPEA
TYPE	WSC or WSCE	WSC or WSCE	WSC or WSCE	WSC or WSCE	3411	3430
MODEL	*NONE	*NONE	*NONE	*NONE	1, 2, or 3	A01
CTLADR	0030	0070	00B0	00F0	0015	0052
DTACPR	N/A	N/A	N/A	N/A	N/A	See note

Note: If hardware data compression feature is installed, DTACPR parameter value must be *YES. If hardware data compression feature is not installed, must be *NO.

DEVICE DESCRIPTIONS

The following chart omits parameters for which the default is taken.

CRTDEVD Command Parameters	System Console	Diskette Magazine Drive	System Printers		Card Device
DEVD	QCONSOLE	QDKT	QSYSPRT	QSYSPRT2	QCARD96
DEVADR	000002	000012	3262 or 5211 000018 3203 000040	3262 or 5211 000058 3203 000041 or 000040	000019
DEVTYPE	CONS	72MD	3203, 3262 or 5211	3203, 3262, or 5211	5424
MODEL	*NONE	1001	A1, B1, 2, or 5	A1, B1, 2, or 5	A1, A2, K1, K2, or K3
PRTIMG	–	–	QSYSIMAGE.QGPL	QSYSIMAGE.QGPL (see note)	–
TEXT	'System Console'	'Diskette Device'	*BLANK	*BLANK	*BLANK

Note: If QSYSPRT2 has a different print belt or print train, it must have a different print image (PRTIMG parameter). This can be QSYSIMAGE2.QGPL.

CRTDEVD Comman Parameters	Magnetic Tape Drives							
DEVD	QTAPE1	QTAPE2	QTAPE3	QTAPE4	QTAPE5	QTAPE6	QTAPE7	QTAPE8
DEVADR	000015	010015	020015	030015	000052	010052	020052	030052
DEVTYPE	3410	3410	3410	3410	3430	3430	3430	3430
MODEL	See note	See note	See note	See note	A01	B01	B01	B01
CTLU	QTAPE	QTAPE	QTAPE	QTAPE	QTAPEA	QTAPEA	QTAPEA	QTAPEA

Note: MODEL must be same model number as 3411 Control Unit to which 3410 tape drive is attached.

If the second system printer used a different print belt (print image) than the first printer, the entry for the print image would not be found in the appendix. However, for your second printer, the supplied print image has a name of QSYSIMAGE2.

IBM®

ZR30-1020-03

Printed in U.S.A.