**zOS Lab3 - Exercises for Chapter 6 (total 26 marks)**

**Using Job Control Language and System Display and Search Facility**

The exercises teach the following tasks:

* Creating a simple job
* Using ISPF in split screen mode
* Manipulating text in ISPF
* Submitting a job and checking the results
* Creating a PDS member
* Copying a PDS member

**6.14.1 Creating a simple job (8 marks)**

Perform the following steps:

1. From ISPF, navigate to the Data Set List Utility panel and enter *yourid*.JCL in the Dsname Level field (described in an earlier exercise).

2. Enter e (edit) to the left (in the command column) of *yourid*.JCL. Enter s (select) to the left of member JCLTEST. Enter RESet on the editor command line.

3. Notice that only a single JCL line is in the data set, that is, EXEC PGM=IEFBR14. This is a system utility that does not request any input or output and is designed to complete with a successful return code (0). NOTE: If there are two lines, enter character \* after // on the first (JOB) line! Enter SUBMIT or SUB on the command line and press Enter.

4. Enter 1 in response to the following message: IKJ56700A ENTER JOBNAME CHARACTER(S) - The result will be the following message: IKJ56250I JOB *yourid*1(JOB00037) SUBMITTED. Whenever you see three asterisks (\*\*\*), it means there is more data to see. Press Enter to continue. When the job finishes, you should see the following message:

$HASP165 yourid1 ENDED AT ZOSKCTR MAXCC=0 CN(INTERNAL)

5. Add (insert) a new first line in your file that will hold a JOB statement. The JOB statement must precede the EXEC statement. (Hint: Replicate (r) the single EXEC statement, then overwrite the EXEC statement with your JOB statement.) This JOB statement should read:

//*yourid*A JOB 1,'*your name'*,NOTIFY=&SYSUID

If it was already there, now removed the \* after // on the first (JOB) line, then change as follows…

Replace yourid with your user ID, leave the A, replace *your name* with tour name, then submit this JCL and press PF3 to save the file and exit the editor.

6. From the ISPF Primary Option Menu, find SDSF (described in 7.9.5, “Using SDSF” on page 294). You can use the split screen function for a new screen session, giving you one session for the DSLIST and the other for SDSF.

7. In the SDSF menu, enter PREFIX yourid\*, then enter ST (Status Panel). Both jobs that you submitted should be listed. Place S (select) to the left of either job, then page up and down to view the messages produced from the execution.

Copy and paste the SDSF Status display of the two jobs you have just submitted (*yourid*1 and *yourid*A) .

KC03CCAA JOB04765 KC03CCA 1 PRINT A 548

KC03CCA1 JOB04772 KC03CCA 1 PRINT A 554

Press PF3 to exit.

8. Edit JCLTEST again, and insert the following lines at the bottom:

//CREATE DD DSN=yourid.MYTEST,DISP=(NEW,CATLG),

// UNIT=SYSDA,SPACE=(TRK,1)

9. Submit the content of JCLTEST created above. Take note of (copy and paste) the job number here (not for marks – just so you have it for later):

Now press PF3 (save and exit edit), then view the output of this job using SDSF. Notice that you have two jobs with the same jobname. The jobname with the highest JOBID number is the last one that was run.

a. What was the condition code? If it was greater than 0, page down to the bottom of the output listing to locate the JCL error message. Correct the JCLTEST and resubmit. Repeat until cond code=0000 is received. Copy and paste the text from your JCLTEST member.

000001 //KC03CCA JOB 1,'YUHSUAN HUANG',NOTIFY=&SYSUID

000002 //STEP1 EXEC PGM=IEFBR14

000003 //CREATE DD DSN=KC03CCA.MYTEST,DISP=(NEW,CATLG),

000004 // UNIT=SYSDA,SPACE=(TRK,1)

b. Navigate to the Data Set List Utility panel (=3.4) and enter *yourid*.MYTEST in the DSNAME level field. What volume was used to store the data set?

KC03CCA.MYTEST KCTR08

c. Enter DEL / in the numbered left (command) column of the data set to delete the data set. A confirmation message may appear asking you to confirm that you want to delete the data set.

d. We just learned that batch execution of program IEFBR14, which requires no inputs or outputs, returns a condition code 0 (success) if there were no JCL errors. Although IEFBR14 does no I/O, JCL instructions are read and executed by the system. This program is useful for creating (DISP=NEW) and deleting (DISP=(OLD,DELETE)) data sets on a DD statement.

10. From any ISPF panel, enter the following command in the Command Field

==>: TSO SUBMIT JCL(JCLERROR)

Your user ID is the prefix (high-level qualifier) of data set JCL containing member JCLERROR.

a. If there were no job card, you would be prompted to enter a suffix character for a generated job card. Take note of the jobname and job number (not for marks) from the submit messages.

b. Use SDSF and select the job output. Since some of the jobs have names other than *yourid*\*, you may find it easier now to filter by owner than by prefix. Page down to the bottom. Do you see the JCL error? What are the incorrect and correct JCL DD operands?

Correct the JCL error located in *yourid*.JCL(JCLERROR). Resubmit JCLERROR to validate your correction.

Copy and paste the corrected JCLERROR member.

000001 //KC03CCA JOB 1,'YUHSUAN HUANG',NOTIFY=&SYSUID

000002 //STEP1 EXEC PGM=IEFBR14

000003 //ALLOC DD DSN=SYS1.PROCLIB,DISP=SHR

11. From any ISPF panel, enter TSO SUBMIT JCL(SORT). Your user ID is the assumed prefix of data set JCL containing member SORT.

a. If there were no job card, you would be prompted to enter a suffix character for a generated job card. Take note of the jobname and job number from the submit messages (not for marks).

b. Use SDSF and place a ? to the left of the job name. The individual listing from the job will be displayed. Place s (select) to the left of SORTOUT to view the sort output.

Copy and paste the first few lines of SORTOUT.

201 NJ

202 DC

203 CT

204 Manitoba

205 AL

206 WA

207 ME

208 ID

209 CA

210 TX

212 NY

213 CA

214 TX

215 PA

216 OH

217 IL

Now press PF3 to return. Select JESJCL. If there were no job card, you would see a “job statement generated message”. Note the “substitution JCL” messages.

12. Purge some (or all) unnecessary job output. From SDSF, place a p (purge) to the left of any job that you would like to purge from the JES output queue.

13. From the ISPF panel, enter TSO SUBMIT JCL(SORT) and review the output.

14. From the ISPF panel, enter TSO SUBMIT JCL(SORTPROC) and review the output. You may not see the output in the SDSF ST panel, because the jobname is not starting with *yourid*. To see all output, enter PRE \*, then OWNER *yourid* to see only the jobs that are owned by you.

What JCL differences exist between SORT and SORTPROC?

The SORTPROC JCL which includes the SORT code inside the procedure. Besides, this SORTPROC file also override the MYSORT.SORTOUT statement.

000001 //KC03CCA JOB 1,'YUHSUAN HUANG',NOTIFY=&SYSUID

000002 //\*---------------------------\*/

000003 //MYPROC PROC

000004 //MYSORT EXEC PGM=SORT

000005 //SYSOUT DD SYSOUT=\*

000006 //SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(5,1))

000007 //SORTOUT DD SYSOUT=\*

000008 //SORTIN DD DISP=SHR,DSN=&SORTDSN

000009 // PEND

000010 //\*---------------------------\*/

000011 //STEP1 EXEC MYPROC,SORTDSN=&SYSUID..JCL(AREACODE)

000012 //MYSORT.SORTOUT DD DSN=&SYSUID..SORT.OUTPUT,

000013 // DISP=(NEW,CATLG),SPACE=(CYL,(1,1)),UNIT=SYSDA,

000014 // DCB=(LRECL=20,BLKSIZE=0,RECFM=FB,DSORG=PS)

000015 //SYSIN DD DISP=SHR,DSN=&SYSUID..JCL(SORTCNTL)

15. In both JCL streams, the SYSIN DD statement references the sort control statement.

Where is the sort control statement located in the SORTPROC JCL?.

000015 //SYSIN DD DISP=SHR,DSN=&SYSUID..JCL(SORTCNTL)

It means this SORTCONT statement located in myid.JCL(SORTCNTL).

16. Edit the partitioned data set member containing the SORT control statement.Change FIELDS=(1,3,CH,A) to FIELDS=(6,20,CH,A). Press PF3 and then from the ISPF panel enter TSO SUBMIT JCL(SORT). Review the job’s output using SDSF.

Was this sorted by code or area?

In the SORTOUT, I can see the output is sort by area.

780 Alberta

403 Alberta

264 Anguilla

268 Antigua/Barbuda

907 AK

205 AL

334 AL

256 AL

251 AL

501 AR

870 AR

479 AR

480 AZ

602 AZ

17. From the ISPF panel, enter TSO LISTC ALL. By default, this command will list all catalog entries for data sets beginning with *yourid*. The system catalog will return the data set names, the name of the catalog storing the detailed information, the volume location, and a devtype number that equates to specific values for JCL UNIT= operand. LISTC is an abbreviation for LISTCAT.

**6.14.2 Using ISPF in split screen mode (3 marks)**

As discussed earlier, most ISPF users favor a split screen. This is easily accomplished by performing the following steps:

1. Move the cursor to the top (or bottom) line.

2. Press PF2 to split the screen.

3. Press PF9 to switch between the two screens.

4. Press PF3 (perhaps several times) to exit from one of the splits.

The screen need not be split at the top or bottom. The split line can be positioned on any

line by using PF2. More than two screens can be used.

Try using these ISPF commands:

START

SWAP LIST

SWAP <screen number.>

Briefly describe what each of these commands does (3 marks).

Typing the “START” command will give us a new split-screen. This one is as same as the Press PF2 to split the screen step.

When I type the “SWAP LIST” command, it will show a pop window called ISPF Task List. I can clearly understand the check which split screen I look at. In addition, I can use the 's' command beside the ID to change to a different split-screen.

Or, I can type “SWAP <screen number>” to swap the split-screen. The screen number is also displayed on the ISPF Task List window.

**6.14.3 Manipulating text in ISPF (4 marks)**

After logging on to TSO/E and activating ISPF, look at the primary option menu and select the appropriate option to edit a data set and perform the following:

1. Create a test member in a partitioned data set (*'yourid*.JCL'). Enter some lines of information, then experiment with each of the commands below. Use PF1 if you need help.

**i** Insert a line.

**Enter key** Press Enter without entering anything to escape insert mode.

**i5** Obtain five input lines.

**d** Delete a line.

**d5** Delete five lines.

**dd/dd** Delete a block of lines (place a DD on the first line of the block and another DD on the last line of the block).

**r** Repeat (or replicate) a line.

**rr/rr** Repeat (replicate) a block of lines (where an RR marks the first line of the block and another RR

marks the last line).

**c along with a or b** Copy a line after or before another line.

**c5 along with a or b** Copy five lines after or before another line.

**cc/cc along with a or b** Copy a block of lines after or before another line.

**m, m5, mm/mm** Move line(s).

**x, x5, xx/xx** Exclude lines.

**s** Redisplay (show) the lines you excluded.

**(** Shift right columns.

**)** Shift left columns.

Copy and paste the contents of your new test member here, including the top line showing the member name (4 marks).

EDIT KC03CCA.JCL(TEST3) - 01.00 Columns 00001 00072

Command ===> Scroll ===> HALF

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Top of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

000001 Hello, this test file is for lab 3 - 6.14.3

000002 My name is YuHsuan Huang

000003 123

000004 456

000005 789

000006 split line

000007 abc

000008 def

000009 ghi

000010 bottom line

- - - - - - - - - - - - - - - - - - - 1 Line(s) not Displayed

000012 bottom line

- - - - - - - - - - - - - - - - - - - 2 Line(s) not Displayed

000015 ttom line

000016 def

000017

000018 def

000019 f

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bottom of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**6.14.4 Submitting a job and checking the results (3 marks)**

Edit member COBOL1 in the *yourid*.LIB.SOURCE library and inspect the COBOL program. If the COBOL1 is missing from your LIB.SOURCE dataset, copy all members from 'ZPROF.ZSCHOLAR.LIB.SOURCE.BIS(\*)' to your 'yourid.LIB.SOURCE' data set.

You may also need to create a data set named 'yourid.PROGRAM.LOAD' if it does not exist. To do this copy (=3.3) 'KC02315.PROGRAM.LOAD(COBOL1)' and specify 'yourid.PROGRAM.LOAD' as the data set to copy to. If the dataset does not exist you will be prompted to allocate the data set using the attributes of ) 'KC02315.PROGRAM.LOAD'.

There is no JCL with it. Now edit member COBOL1 in *yourid*.JCL. Inspect the JCL carefully. It uses a JCL procedure to compile and run a COBOL program. Perform the following steps:

1. Change the job name to *yourid* plus additional characters.

2. Change the NOTIFY parameter to your user ID.

3. Add TYPRUN=SCAN to your job card.

4. Type SUB on the ISPF command line to submit the job.

5. Split your ISPF screen and go to SDSF on the new screen (you might have this already from an earlier exercise).

6. In SDSF, go to the ST (Status) display and look for your job name. You may need to enter a PRE or OWNER command on the SDSF command line to see any job names. (A previous user may have issued a prefix command to see only certain job names.)

7. Type S beside your job name to see all of the printed output:

– Messages from JES2

– Messages from the initiator

– Messages from the COBOL compiler

– Messages from the binder

– Output from the COBOL program

Did the job actually execute?

No, this job not been executed.

What is displayed?

Is shows ‘3 IEFC001I PROCEDURE IGYWCL WAS EXPANDED USING PRIVATE LIBRARY ZPROF.ZSC’ in the JES2.

8. Remove TYPRUN=SCAN when you are ready to run your job.

9. Display the job's output using SDSF.

What was the 'COND CODE' of STEP1 in your COBOL job?

COND CODE is 0012

14.04.12 JOB06634 $HASP165 KC03CCAN ENDED AT SVSCJES2 MAXCC=0012 CN(INTERNAL)

**6.14.5 Creating a PDS member (4 marks)**

There are several ways to create a new PDS member. Try each of the following, using your own user ID. In the following steps, TEST3, TEST4, TEST5, and TEST6 represent new member names. Enter a few lines of text in each case. Using the ISPF edit panel, perform the following steps:

1. Go to the ISPF primary menu.

2. Go to option 2 (Edit).

3. In the Data Set Name line, enter JCL(TEST3) (no quotes).

4. Enter a few text lines and press PF3 to save the new member.

5. A new member can be created while viewing the member list in edit mode by performing the following steps:

1. Use option 3.4 (or option 2) to edit *yourid*.JCL.

2. While viewing the member list, enter S TEST4 in the command line.

3. Enter a few text lines and press PF3 to save the new member.

6. A new member can be created while editing an existing member by performing the following steps:

1. Edit *yourid*.JCL(TEST1) or any other existing member.

2. Select a block of lines by entering cc (in the line command area) in the first and last lines of the block.

3. Enter CREATE TEST5 on the command line, which creates the TEST5 member in the current library.

7. A new member can be created with JCL. Enter the following JCL in *yourid*.JCL(TEST5) or any other convenient location:

//*yourid*1 JOB 1,JOE,MSGCLASS=X

//STEP1 EXEC PGM=IEBGENER

//SYSIN DD DUMMY

//SYSPRINT DD SYSOUT=\*

//SYSUT2 DD DISP=OLD,DSN=*yourid*.JCL(TEST6)

//SYSUT1 DD \*

This is some text to put in the member

More text

/\*

Save the member, submit the job and rerun until you get a completion code of 0.

Copy and paste the new member list of your 'yourid.JCL' data set (including the new members you have created) (4 marks).

EDIT KC03CCA.JCL Row 0000001 of 0000018

Command ===> Scroll ===> PAGE

Name Prompt Size Created Changed ID

\_\_\_\_\_\_\_\_\_ AREACODE

\_\_\_\_\_\_\_\_\_ COBOL1 6 2008/10/05 2021/02/19 14:03:10 KC03CCA

\_\_\_\_\_\_\_\_\_ COMMENT

\_\_\_\_\_\_\_\_\_ EDITTEST 5 2013/05/21 2021/01/25 16:07:30 KC03CCA

\_\_\_\_\_\_\_\_\_ JCLERROR 3 2010/10/07 2021/02/17 20:50:58 KC03CCA

\_\_\_\_\_\_\_\_\_ JCLTEST 4 2008/10/05 2021/02/17 20:20:44 KC03CCA

\_\_\_\_\_\_\_\_\_ LAB1 7 1999/09/08 1999/09/08 14:25:50 INSTBJH

\_\_\_\_\_\_\_\_\_ SORT 7 2010/10/07 2021/02/17 21:19:22 KC03CCA

\_\_\_\_\_\_\_\_\_ SORTCNTL 1 2011/10/03 2021/02/19 10:59:51 KC03CCA

\_\_\_\_\_\_\_\_\_ SORTPROC 15 2012/10/01 2021/02/17 21:42:01 KC03CCA

\_\_\_\_\_\_\_\_\_ SQLIN 10 1997/12/18 1999/02/22 23:10:56 INSTBJH

\_\_\_\_\_\_\_\_\_ T$E$M

\_\_\_\_\_\_\_\_\_ TEST 4 2021/01/22 2021/01/22 11:04:25 KC03CCA

\_\_\_\_\_\_\_\_\_ TEST2 12 2021/02/03 2021/02/03 20:57:26 KC03CCA

\_\_\_\_\_\_\_\_\_ TEST3 5 2021/02/19 2021/02/19 14:10:46 KC03CCA

\_\_\_\_\_\_\_\_\_ TEST4 5 2021/02/19 2021/02/19 14:12:08 KC03CCA

\_\_\_\_\_\_\_\_\_ TEST5 11 2021/02/19 2021/02/19 14:23:25 KC03CCA

\_\_\_\_\_\_\_\_\_ TEST6

\*\*End\*\*

**6.14.6 Copying a PDS member (4 marks)**

There are many ways to copy a library member. An earlier exercise used the ISPF 3.3 panel function to copy all the members of a library. The same function can be used to copy one or more members.

While editing a library member, we can copy another member of the library into it by performing the following steps:

1. Edit a library member.

Copy and paste the original contents of the member.

EDIT KC03CCA.LIB.SOURCE(COBOL1) - 01.02 Columns 00001 00072

Command ===> Scroll ===> HALF

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Top of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

==MSG> -CAUTION- Profile changed to NUMBER ON STD (from NUMBER OFF).

==MSG> Data has valid standard numbers.

==MSG> -CAUTION- Profile changed to CAPS OFF (from CAPS ON) because data

==MSG> contains lower case characters.

==MSG> -Warning- The UNDO command is not available until you change

==MSG> your edit profile using the command RECOVERY ON.

000100 //COBOL1 JOB 1,NOTIFY=&SYSUID

000200 //COBRUN EXEC IGYWCLG

000300 //COBOL.SYSIN DD \*

000400 IDENTIFICATION DIVISION.

000500 PROGRAM-ID. HELLO.

000600 PROCEDURE DIVISION.

000700 DISPLAY 'hello world.'.

000800 STOP RUN.

000900 /\*

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bottom of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2. Mark a line in this member with a (after) or b (before) to indicate where the other member should be copied.

3. Enter COPY *xxx* on the command line, where *xxx* is the name of another member in the current data set.

Copy and paste the new contents of the member.

EDIT KC03CCA.LIB.SOURCE(COBOL1) - 01.03 Member ALLOC copied

Command ===> Scroll ===> HALF

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Top of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

==MSG> -CAUTION- Profile changed to NUMBER ON STD (from NUMBER OFF).

==MSG> Data has valid standard numbers.

==MSG> -CAUTION- Profile changed to CAPS OFF (from CAPS ON) because data

==MSG> contains lower case characters.

==MSG> -Warning- The UNDO command is not available until you change

==MSG> your edit profile using the command RECOVERY ON.

000100 //COBOL1 JOB 1,NOTIFY=&SYSUID

000200 //COBRUN EXEC IGYWCLG

000300 //COBOL.SYSIN DD \*

000400 IDENTIFICATION DIVISION.

000500 PROGRAM-ID. HELLO.

000510 //ALLOCATE JOB 1

000520 // EXEC PGM=IEFBR14

000530 //Z DD DSN=ZSCHOLAR.JCL,DISP=(,CATLG),

000540 // DCB=(LRECL=80,RECFM=FB,DSORG=PO),

000550 // SPACE=(CYL,(2,1,25)),UNIT=SYSDA

000600 PROCEDURE DIVISION.

000700 DISPLAY 'hello world.'.

000800 STOP RUN.

000900 /\*

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bottom of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

We can copy a member from another data set (or a sequential data set) by performing the following steps:

1. Edit a member or sequential data set.

Copy and paste the original contents of the member or data set.

EDIT KC03CCA.JCL(TEST4) - 01.00 Columns 00001 00072

Command ===> Scroll ===> HALF

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Top of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

==MSG> -Warning- The UNDO command is not available until you change

==MSG> your edit profile using the command RECOVERY ON.

000001 Hey, test file for lab 3 - 6.14.5

000002 My name 'YuHsuan Huang'

000003 123

000004 321

000005 1234567

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Bottom of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2. Mark a line with A (after) or B (before) to indicate where to insert the new material.

3. Enter COPY on the command line to display the Edit/View-Copy panel.

4. Enter the full sequential data set name (with single quotes, if necessary) or a full library name (including member name) in the Data Set Name field.

Copy and paste the new contents of the member or data set.

Copy member from KC03CCA.LIB.SOURCE(HELLO)’.

EDIT KC03CCA.JCL(TEST4) - 01.01 Columns 00001 00072

Command ===> Scroll ===> HALF

\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Top of Data \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

==MSG> -Warning- The UNDO command is not available until you change

==MSG> your edit profile using the command RECOVERY ON.

000001 Hey, test file for lab 3 - 6.14.5

000002 My name 'YuHsuan Huang'

000003 123

000004 321

000005 1234567

000006 //HELLO JOB (POK,KC02014),MGELINSKI,MSGLEVEL=(1,1),MSGCLASS=X,

000007 // CLASS=A,NOTIFY=&SYSUID

000008 /\*JOBPARM SYSAFF=\*

000009 // JCLLIB ORDER=(IGY.SIGYPROC)

000010 //\*

000011 //RUNIVP EXEC IGYWCLG,PARM.COBOL=RENT,REGION=1400K,

000012 // PARM.LKED='LIST,XREF,LET,MAP'

000013 //COBOL.STEPLIB DD DSN=IGY.SIGYCOMP,

000014 // DISP=SHR

000015 //COBOL.SYSIN DD \*

000016 IDENTIFICATION DIVISION.

000017 PROGRAM-ID. CALLIVP1.

000018 AUTHOR. IBM PROGRAMMER.

000019 INSTALLATION. ITSO

000020 DATE-WRITTEN. JUL 27, 2004.

000021

000022