**CSCI 360 Assignment 1: First Mainframe Program**

**25 points**

**To Start**

First, after signing on to Marist using TSO/ISPF as shown to you in class, allocate your assignments PDSE that you will use all semester. In the paragraphs that follow, please name this data set exactly as follows:

KC03nnn.CSCI360.ASSIGNS

In the name, KC03nnn is your Marist ID as assigned to you in the document named *Marist IDs* in Blackboard's *Course Documents*. By naming your data sets as instructed here enables the TA and instructor to review your progress and, when necessary, to quickly and efficiently copy PDSE members in order to assist you in debugging and/or correcting mistakes in assignments.

Use ISPF Option 3.3 to allocate your ASSIGNS PDSE with space units of TRKS (tracks), a primary quantity of 10, a secondary quantity of 10, 5 directory blocks, record format of FB (fixed blocked), a record length of 80 (bytes), and a block size of 880, and, near the bottom of the screen, a data set name type of LIBRARY.

Note that this is all covered in the first document, *360 1. Allocating your Assignment PDSE*, in folder **Using ISPF on the Marist Mainframe** found in *Notes and Slides* on Blackboard.

**YOU are responsible for backing up this very important data set on a regular basis, i.e., frequently.**

**The Assignment**

After your ASSIGNS PDSE is allocated, type or copy and paste the program below into the editor. Save the member as ASSIGN1. Submit the job and review your output in ISPF Option SD.ST. Please note that EACH assignment you write this semester, name the PDSE member for that assignment appropriately, such as, for example, ASSIGN1, ASSIGN2, ASSIGN3, ASSIGN5A, ASSIGN5B, etc.

You do NOT need to type all of the documentation shown below into your member but you must include the first five lines beginning with CSCI 360 and ending with DUE DATE plus the closing line of asterisks.

Be sure to change the *nnn* in the first line of the JCL to reflect the last three characters of your Marist ID, change *your first initial and last name* in the first line of the JCL to your first initial followed by a period and a space and your last name and, in the program documentation box, change -*n* to your section number*, current semester* to the current semester like FALL 2024, *your full name* to your own full name, update the due date of the assignment, fill in the only, or driver, CSECT name and the program's function. In future assignments, you will change the assignment number and fill in the other information as necessary.

For more information about coding and documentation, refer to the document titled *360 Coding and Documentation Guidelines.pdf* found in *Course Documents* on Blackboard.

//KC03*nnn*A JOB ,'*your first initial and last name*',MSGCLASS=H

//JSTEP01 EXEC PGM=ASSIST

//STEPLIB DD DSN=KC00NIU.ASSIST.LOADLIB,DISP=SHR

//SYSPRINT DD SYSOUT=\*

//SYSIN DD \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \*

\* CSCI 360-*n* ASSIGNMENT 1 *current semester* \*

\* \*

\* PROGRAMMER NAME: *your full name*  \*

\* DATE DUE: *due date of assignment* \*

\* \*

\* MAIN PROGRAM NAME: *The name of the only, or driver, CSECT* \* \* \*

\* FUNCTION: The function of this program is to \*

\* provide an example of a simple ASSIST \*

\* Assembler program and documentation. \*

\* \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

PROGRAM1 CSECT

USING PROGRAM1,15 ESTABLISH ADDRESSABILITY ON REG 15

LA 4,13 LOAD 13 INTO REG 4

LA 8,6 LOAD 6 INTO REG 8

AR 4,8 ADD REG 8'S CONTENTS TO REG 4'S

XDUMP , DUMP CONTENTS OF ALL 16 REGS

BCR B'1111',14 UNCONDITIONAL RETURN TO CALLER (OS)

\*

LTORG LITERAL ORGANIZATION

\*

END PROGRAM1

/\*

//

It is very important that you not save any blank lines before or after what is presented above. The columns on each line and where things appear here are also very important.

In the Assembler language, columns 1, 10 and 16 are significant. Labels like PROGRAM1 begin in column 1, instruction mnemonics like LA (Load Address) begin in column 10 and instruction operands like 3,12 begin in column 16. All letters must be capital letters too!

Assembler language, or "Assembler", is a very unforgiving language that requires close attention to detail. The lines of the job above that begin with at least one forward slash (/) are lines of JCL, or Job Control Language, and are ***not*** Assembler code.

We will use the same JCL for all our programming assignments and in any examples for the remainder of the semester. There will be minor changes with lines of information added to the end of the job but that will come later.

The JCL above surrounds the Assembler program and is used on the mainframe to tell the operating system, z/OS, what we are trying to accomplish. In this case, we are compiling an Assembly language program and, if that is successful, we want to execute, or "run", it. We do not call it compiling, though, when using the Assembler; it is called "assembling" rather than "compiling".

In summary, the first line of the JCL tells the mainframe operating system, z/OS, that it is **your** job with your Marist ID. The word JOB tells the mainframe operating system, z/OS, that the first line is the beginning of a new "job." Note the required space between JOB and the comma. Inside the single quotes (apostrophes which mainframers call "ticks" or "tick marks"), you can put up to 20 characters of your choice. For this class, please put your first and last name or, if too long, your first initial, a period, a space and your last name. The MSGCLASS=H tells the Marist system where to place your job's output when it finishes. H is the "held" class and your job's output will be placed in the held output queue.

The line with EXEC PGM=ASSIST tells the Marist system that you want to execute the program object, or executable, named ASSIST. A program object is what we call an "executable" on the mainframe. (You will also hear executables called "load modules" but it is old terminology.) ASSIST is a learning version of the mainframe Assembler that assembles, or "compiles," your program and actually executes it too but only if it assembles with no errors.

When you are done typing and you are ready to run your job and view the results, first save your work and then type SUB on the command line and press ENTER.

After you submit the job, you will see the ***job number*** assigned by JES (the Job Entry Subsystem) displayed. Press ENTER to get rid of the job number being displayed. Then type =SD.ST on the command line and press ENTER.

When you are satisfied that the results are correct, you must get a copy of your output down from Marist onto your own machine and convert it into a text document named ASSIGN1.txt (change the name to ASSIGN1.txt, if necessary) to submit it on Blackboard. You will be shown how to do this within the first few days of class using the Retrieve Jobs website which not only downloads your Marist output but also converts it to a .txt file. Make sure the downloaded file is named ASSIGN1.txt for this assignment, to ASSIGN2.txt for the next assignment, etc.

Once you have the text document on your own machine, it is VERY important that you do NOT change anything **in** the text document you plan to submit. **If you do, it is considered cheating!**

Submit a copy of your ASSIGN1.txt file on Blackboard by the time and date it is due. Also, copy your ASSIGN1 member from your ASSIGNS PDSE into your SUBMTTD PDSE and be sure you do not open and/or change it! The date and time stamp of the member of the SUBMTTD PDSE are important.

© 2024 Geoffrey D. Decker