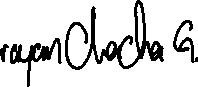
**CSCI 360 Assignment 4 Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**



**Summer 2020 ABENDs and Dump Reading**

**25 points**

**This programming assignment does not require any further coding or documentation than what is provided. The program will ABEND and your task is to learn how to investigate what happened so that you can debug your own Assembler programs in the future. To begin, run the following program on the Marist mainframe using the ASSIST JCL used previously. Be sure you type it or copy it EXACTLY as shown below:**

DUMPEX5 CSECT

USING DUMPEX5,15 ESTABLISH REG 15 AS BASE REG

\*

SR 3,3 CLEAR REG 3

\*

LA 5,VAR1 LOAD ADDRESS OF VAR1 INTO REG 5

LA 6,VAR2 LOAD ADDRESS OF VAR2 INTO REG 6

\*

A 3,0(,5) ADD VAR1 TO REG 3

A 3,0(,6) ADD VAR2 TO REG 3

\*

LA 8,ADDED LOAD ADDRESS OF VARIABLE ADDED INTO REG 8

ST 3,0(,8) STORE THE SUM OF VAR1 & VAR2 AT ADDED

\*

XDUMP , DUMP REGS

\*

LTORG

\*

ADDED DC F'0' FULLWORD OF ZERO TO HOLD SUM

\*

VAR1 DC F'1206000000' VARIOUS FULLWORD INTEGER VALUES

VAR2 DC F'972460'

VAR3 DC F'1344335922'

\*

END DUMPEX5

**Use the resulting ABEND dump output to answer the questions below. Each is worth 2 points except question 11 which is worth 5 points.**

1. Did this error occur (a) while the program was being assembled or (b) when it was being run?

**(b) while it was being run**

1. What is the address of the next instruction which would have been executed?

**000030**

1. What is the value of the condition code at the time of the ABEND?

**00**

1. What is the length of the instruction that caused the ABEND (number of bytes)?

**2 remainder 1 (hex), 2 remainder 1(decimal), The ABEND instruction is 4 bytes long**

1. What is the address of the instruction that caused the abend?

**000030 – 4 = 00002C**

1. What type of error occurred (number and name)?

**0006(hex) Specification Exception**

1. What usually causes this error?

**This error code pretty much is caused by the format address of D(B), or D(X,B) resolved to a value that is not on the boundary needed for the instruction involve.**

1. What does the value in register 3 represent at the time of the ABEND dump?

**What register 3 represent at the time of the ABEND dump is the storage of the sum of VAR1 and VAR2 added together.**

1. What instruction needs to be added to fix this ABEND?

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

1. What does the value stored at location counter value 00002C represent?

**The location counter value of** 00002C **determines which instruction caused the ABEND. It makes it easier for us as the developers to identify where is the location of where an error might have taken place.**

1. What exactly happened here to cause the ABEND? Be detailed but succinct in your description.

**What happen, after copying the code I got the error code (SYSTEM = 0C6 SPECIFICATION, 0006). I at first I went to the address of were the error was being caused. I noticed that VAR3 was not doing much, it was just a label that stored the value (1344335922).That trigger my mind, because the format of VAR3 was correct, so that wasn’t the problem. What caused this ABEND was that the CPU was looking for more instructions to execute and interpret. Even though VAR3 wasn’t an instruction the CPU doesn’t know that. So by adding B’1111’,14 this will cause the CC to only be 0 and 3 and all four bits in the mask are 1, so by adding this instruction this caused the program to jump to the address in register 14 (unconditional Branch) and that fixed the ABEND.**