**CSCI 360-1 Final Exam Study Questions Spring 2019**

**Part I. Multiple Choice**

**Read each statement/question carefully and thoroughly. Write the letter of the answer that is most correct in the space provided.**

1. On entry to an external subroutine, register 13 contains the address of the calling routine’s register save area.
2. True
3. False
4. The PACK instruction can cause a data exception.
5. True
6. False
7. A S0C5 is caused when a program tries to access an address outside of an actual memory location.
8. True
9. False
10. Hexadecimal X'67' is represented in binary as B'01100111'.
11. True
12. False
13. The address of a fullword in storage must end with:
14. 0, 2, 4, 6, 8, B, or D
15. 0, 4, 8, or C
16. 0, 2, 4, 6, 8, A, C, or E
17. 0, 2, 4, or 8
18. None of the above.
19. The save area in standard linkage conventions is:
20. 16 bytes
21. 18 bytes
22. 64 bytes
23. 72 bytes
24. None of the above.
25. How many bits are in a byte?
26. 2
27. 4
28. 8
29. 12
30. None of the above.
31. A doubleword is 8 bits long
32. True
33. False
34. Which of the following is the most efficient instruction to multiply the contents of register 5 by 40?
35. M 5,=F'40'
36. M 5,40
37. M 4,40
38. M 4,=F'40'
39. None of the above.
40. When the extended mnemonic BNH is coded, when will the branch be taken?
41. Only when the first operand is greater than the second operand.
42. Only when the second operand is greater than the first operand.
43. When the first operand is greater than or equal to the second operand.
44. When the second operand is greater than or equal to the first operand.
45. None of the above.
46. What will be the amount and direction of the shift in the following SRP instruction?

SRP FIELD(5),60,0

1. Right shift – 4 digits
2. Left shift – 4 digits
3. Right shift – 6 digits
4. Left shift – 6 digits
5. None of the above.
6. The instruction in question 11 would round a result.
7. True
8. False
9. L 5,=F'-10' will place \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ in register 5.
10. 0000FFF6
11. FFFFFFF6
12. 0000FFFF
13. FFFFFFFF
14. None of the above.
15. When using standard linkage conventions, what can a called program always assume upon entry?
16. Register 14 contains the address of the called routine.
17. Register 15 contains the address of the next line of code to execute in the calling routine.
18. Register 13 contains the address of the register save area in the calling routine.
19. Register 1 contains the address of the input area.
20. None of the above.
21. On entry to an external routine, register 15 contains \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
22. the address of the calling routine save area
23. the address of the first instruction in the called routine
24. the address of the called routine’s save area
25. the address of the parameter list
26. None of the above.
27. The first line of standard entry linkage, STM 14,12,12(13), stores registers in the \_\_\_\_\_\_\_\_\_\_\_.
28. calling routine save area
29. called routine save area
30. cache memory
31. storage following the LTORG
32. None of the above.
33. On entry to an external routine, register 14 contains \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
34. the address of the calling routine save area
35. the address of the first instruction in the called routine
36. the address of the called routine’s save area
37. the address of the next instruction to execute in the calling routine
38. None of the above.
39. On entry to an external routine, register 1 contains \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
40. the address of the calling routine save area
41. the address of the first instruction in the called routine
42. the address of the called routine’s save area
43. the address of the parameter list
44. None of the above.

**Part III. Reading Storage**

**Use the following REGISTER CONTENTS/STORAGE for answering questions 19 and 20. Each question is independent, i.e., the outcome of one will NOT affect the other questions.**

REGS 0-7 00000051 FF0000A4 FF000000 00000038 00000018 00000005 00000016 00000005

REGS 8-15 F4F4F4F4 F4F4F4F4 00000020 F4F4F4F4 00000028 00000118 FFFE7960 00000000

USER STORAGE

000000 4120F0A0 1B33E000 00230050 4750F056 5340F050 50423000 41650004 E000F050

000020 000F191E F00C1B33 4170F065 4150F118 F510F5F5 F5F5F5F5 17202D1D F5F5F5F5

1. What will be the contents of the receiving field after execution of:

LA 3,20(1,0)

1. FF0000B8
2. 000000B8
3. 00000109
4. 000000C4
5. None of the above.
6. What will be the contents of the receiving field after execution of:

L 3,10(7,5)

1. 00000014
2. 50423000
3. F0654150
4. 000F191E
5. None of the above.
6. The following will cause a S0C6:

ST 2,46(,15)

1. True
2. False

# Part II. Dump Reading

Use the ASSIST DUMP that is given below to answer questions 20 through 26. Note that, if you find instructions involved that we have not covered in class, do not worry about it. But, if asked to find the new instruction encoded in storage, you ***should*** be able to answer the question.

PSW AT ABEND FFC50007 E0000018

REG 0-7 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4

REG 8-15 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 00000030 FFFE7960 00000000

USER STORAGE

000000 F830F01A F028F224 F015F023 FA32F01A F01EFB31 F01AF021 07FE0000 098C0009

000020 8CF5F5F0 F0F0F9C8 0CF5F5F5 F5F5F5F5 F5F5F5F5 00000000 F5F5F5F5 F5F5F5F5

000040 F5F5F5F5 F5F5F5F5 F5F5F5F5 F5F5F5F5 F5F5F5F5 F5F5F5F5 F5F5F5F5 F5F5F5F5

1. What is the address of the instruction that WOULD HAVE BEEN EXECUTED NEXT had the program not ABENDed?
2. What is the length (in bytes) of the instruction that caused the ABEND?
3. What is the address of the instruction that caused the ABEND?
4. What type of program interrupt occurred? GIVE THE NAME not just the number.
5. Give the machine code for the instruction that caused the ABEND.
6. Give the ASSEMBLER instruction that caused the ABEND.
7. Why did the ABEND occur? In order to receive full credit, make your answer as specific as possible.

**Part III. Short Coding**

**Give an instruction or instructions that will accomplish each of the following tasks.**

1. Write a loop that will count the number of records in an input file. Assume that a storage area called INBUF has been properly declared.
2. Write the instructions to establish addressability for a DSECT named $TBLREC and then remove its addressability. You may assume the DSECT has already been defined properly.
3. A byte is \_\_\_\_\_ bits in length. A halfword is \_\_\_\_\_ bytes or \_\_\_\_\_ bits in length. A fullword is \_\_\_\_\_ bytes or \_\_\_\_\_ bits in length. A doubleword is \_\_\_\_\_ fullwords, \_\_\_\_\_ bytes, or \_\_\_\_\_ bits in length. A fullword constant pads on the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ with \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. A doubleword constant pads on the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. A character constant pads on the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ with \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.