2021 Fall CPSC 240-1 Answers

Midterm Concepts Test #1

28 Sept 2021: 2:30pm – 6:30pm

Read me

Place your answers in the space following each question.

Place your name in the test either in this front page or on the last page or both.

Near 6:25pm begin to save your test document in either odt, or doc, or docx format.

Before 6:29pm send your document as an attachment to me: holliday@fullerton.edu

If you encounter a question where you feel that you must guess an answer then place the word “Blank” in the space for the answer and you will receive 20% of the credit for that question.

If the answer space is empty then the points for that question are zero.

You may use any word processing tool at your disposal provided it can save files in one of the three accepted formats.

If your computer has no word processor program then try Google Docs, which saves files in every format ever created on this planet.

The total point value of this test is 100 points, which is one-sixth of your course grade.

Every effort has been made to create unambiguous questions. If a question is truly ambiguous send me ordinary email or a chat message to ask for clarification. I will be at computer during the test period probably answering the backlog of email.

This is an open note test. Use any tools that work for you such as:

- recordings of lectures
- class notes
- ebooks
- calculators
- internet

Proceed to the next page.

1. What are the properties that characterize a Von Neumann computer. [3]

Answer: The computer must have all of these components:

CPU	
Primary storage	aka RAM or memory
Secondary storage	aka HDD or SSD or Nvme
I/O devices	such as mouse, keyboard, monitor
Bus	that is, cables for transmission of data

2. What is the model of the oldest microprocessor in the x86 family? This one is so old that some online lists fail to recognize it. [3]

Answer: Intel model 4004

3. What is an assembly instruction that will copy the second quad word from the top of the system stack into the low half of xmm12? [3]

Answer: movsd xmm12, [rsp+8]

4. Write an explanation for a freshman majoring in Philosophy what an MSB is. [3]

Hello Philosophy freshman,

Let's suppose you graduated and went to work at Acme Philosophy Company. Instead of using direct deposit they paid you with a stack of paper money. You carry your stack of US currency back to your work station where you sort the stack in order with $100 notes on top, $50 notes below those, and $20 notes below the 50's, and so to $1 notes on the bottom.

If a big draft of wind came along and blew away some of your currency, which end of the stack do you prefer to remain in spite of the wind.

Freshman says, "That's easy. I prefer the 100's to remain even if I lost a few of the ones.".

You say: "That's right. You kept the MSB because that's the most important part of your large salary".

Freshman says, "Oh, I see. The MSB is the part of the number that counts for the most".

You say: "You're welcome."

==========

Other narratives are possible. The goal is to explain a technical subject to a non-technical person. Answers to this question will be evaluated on an individual basis.

5. Show assembly instruction(s) that will convert 5 into 5.0 and place the 5.0 into xmm5? [3]

Answer:
```
mov rax, 5
cvtsi2sd xmm5, rax
```

//This is one correct answer. There may be others. Each answer submitted will be evaluated for correctness.

6. This integer is written in big endian: 0x2945 10FA 639B 3E7D. Compute the negative of this number and write that negative in little endian. [4]

Answer: First replace each digit with its complement. Then we have

0xD6BA EF05 9C64 C182

Next add 1, and the answer is 0xD6BA EF05 9C64 C183

The last step is to invert the bytes: 0x83C1 649C 05EF BAD6

7. Suppose you are working with 4 byte registers. Convert this decimal integer 2187 to twos complement hex in big endian. [7]

Answer: Let's use repeated division.
2187/2 = 1093 R 1

1093/2 = 546 R 1

546/2 = 273 R 0

273/2 = 136 R 1

136/2 = 68 R 0

68/2 = 34 R 0

34/2 = 17 R 0

17/2 = 8 R 1

8/2 = 4 R 0

4/2 = 2 R 0

2/2 = 1 R 0

1/2 = 0 R 1

0/2 = 0 R 0

Time to stop

100010001011, but we need 32 bits. So far we have only 12 bits. So add 20 leading zeros.

0000 0000 0000 0000 0000 1000 1000 1011. For better readability we write this number in hex.

0X0000 088B

8. Suppose you wrote a term paper for POSC100 class titled “Rights of a Juror”mean. Your teacher said it was a marvelous piece of research. Now you want to post it online for others to view. What license should be placed on this work of intellectual property? [4]

Answer: Creative Commons

9. Let’s suppose you invented new kind of sort function that executed in O(n) time. It was faster than any previously known sort function. You decided to make it public by posting the source code online. What license should you put on that software? [4]

Answer: LGPL which means Lesser General Public License

10. What is the purpose of the “-c” parameter sometimes found in some g++ and gcc compilation statements? [4]

Answer: It means create an object file, but do not attempt to link anything.

In brief, “-c” means compile only.

The default action, without the “-c”, is compile and link in a single step.

11. Convert the number 737.1875 to binary scientific notation. [7]

Answer: 737 1(512)+0(256)+1(128)+1(64)+1(32)+0(16+0(8)+0(4)+0(2)+1(1)

737 = 1011100001

```
 737
-512
 225
-128
  97
 -64
  33
 -32
   1
```

Now work on the fractional part:

0.1875 x 2 = 0.375
0.375 x 2 = 0.75
0.75 x 2 = 1.5
0.5 x 2 = 1.0
0.0 x 2 = 0/0
Time to stop. The fraction part is 0.1875 = 0.0011

Now combine the whole number and the fraction number:

737.1875 = 1011100001.0011 x 2^0 = 1.0111000010011 x 2^9. That is good enough.

12. Convert the number 1.11001 x 2^-3 to base 10 number. [7]

Answer: 1.11001 = 1 + (½) + (¼) + (1/32)

= 57/32 [Add all the fraction over a common denominator]

That is a partial answer. We want the following:

1.11001 x 2^-3 = 57/32 x 2^-3 = (57/32) x (1/8) = 57/256 That's not a pretty fraction, but that is what the answer is.

Use a calculator and the answer becomes: 0.22265625

13. Suppose r12 and r13 both hold large integers. Show assembly statements that will divide r12 by r13 and place the quotient in r14 and the remainder in r15. [7]

Answer:

```
mov  rax,  r12
cqo
idiv  r13
mov  r14,  rax
mov  r15,  rdx
```

14. Show how to zero out an xmm register at high speed. [4]

Answer; xorpd xmm4, xmm4

//Notes: The instruction xorsd does not exist, hence we must use xorpd

//The companion operation for GPRs is xor rcx,rcx

15. Comparison: registers xmm8 and xmm9 both hold valid float numbers. Show asm instruction that will accomplish the following: the arrow means assignment. [7]

```
if xmm8 > xmm9 then
        r8 ← 1
else if xmm8 < xmm9 then
        r9 ← 1
else
        {r8 ← 0
         r9 ← 0}
```

Answer

```
ucomisd  xmm8, xmm9
ja  first_then
jb  second_then
    mov r8, 0
    mov r9,0
    jmp continue

first_then:
    mov r8, 1
    jmp continue

second_then
    mov r9, 1

continue:
```

//Notes: There are 3 possible outcomes: xmm8 > xmm9, xmm8 < xmm9, xmm8 == xmm9

Hopefully, you can match the outcomes with the color highlighting above.

GDB questions. Assume that for each question a program is running in GDB mode. In each case find the command that satisfies the request.

Declaration: An array is declared in C++: long a[7] = {6,15,-1,14,-8,9,3};

16. Show all the values stored in a in hex. [3]

Answer p/x a

17. Show the value stored at index 2 in unsigned integer. [3]

Answer: p/u a[2]

//To everyone: Some of these GDB questions have more than one correct answer. If your test shows another gdb command then test your answer in a live program. If the live program shows that your answer is in fact correct, tell me about your answer and I will gladly restore the lost points. --prof.

Declaration: This array was declared in a C++ file. double b[5] = {3.5,-7.7,8.33, 5.15, 9.1};

18. Show all the numbers of the array in standard floating point form. [3]

Answer: p/f b

19. Show the address where the number 3.5 is stored. [3]

Answer: p/x &b

//Note: The answer above relies on the default index = 0. A complete answer would be
p/x &b[0]

//0 could be replaced by other indices.

Declaration: This variable was declared in a C++ function: double house = 3.55;

20. Show the address where 3.55 is stored. [3]

Answer: p/x &house

21. Show the number stored there in hex. [3]

x/1xg &house

22. What GDB command could have made the following output? [3]

```
0x7fffffffdde0:  0xdf88  0xffff  0x7fff  0x0000  0xff90  0xffff  0x0001  0x0000
0x7fffffffddf0:  0xfc19  0xffff  0xffff  0xffff  0x0000  0x0000  0x8000  0x4017
0x7fffffffde00:  0x0027  0x0000  0x0000  0x0000  0x9913  0xf7c7  0x7fff  0x0000
0x7fffffffde10:  0x0000  0x0000  0x0000  0x0000  0x0000  0x0000  0x0000  x0000
0x7fffffffde20:  0x0000  0x0000  0x0000  0x0000  0x0000  0x0000  0x0000  0x0000
0x7fffffffde30:  0x0280  0x0000  0x0000  0x0000  0x0029  0x0000  0x0000  0x0000
0x7fffffffde40:  0x1c00  0x0001  0x0000  0x0000  0x0079  0x0000  0x608b  0x0000
0x7fffffffde50:  0x000a  0x0000  0x0000  0x0000  0x150d  0x0040  0x0000  0x0000
0x7fffffffde60:  0xefc8  0xf7dc  0x7fff  0x0000  0x14c0  0x0040  0x0000  0x0000
0x7fffffffde70:  0x0000  0x0000  0x0000  0x0000  0x10a0  0x0040  0x0000  0x0000
```

Answer: 10 rows x 8 columns = 80 groups display.
Each group = 4 hex digits = 2 bytes = 1 word = symbol ‘h’

Command is x/80xh $rsp

Suppose this array of char was declared in the section .data

final db “That’s all folks”,10,0

23. Show the starting address of the data followed by entire printable string. [3]

Answer: x/s &final

24. Show the ascii value of each of the first 8 bytes of the array. [3]

Answer: x/8cb &final <==It does in fact answer this question, but the output is not fully satisfactory because the output is a mixture of ascii numeric values and printable chars.

x/s (char[8])final <== Shows the display character of each of the first 8 bytes: incorrect for this question.

25. Show all the printable chars of the array. [3]

Answer: x/s (char[17])final

Wrong: p/x (char[17])final <== Shows only ascii numbers in hex.

//Notes: The answer shown above does produce correct results but it has the horrible requirement that the user know the length of the string before using the command. I have searched extensively for a solution that does not require the length of the string.

Alternate answer: Use the answer from question 23.

Total points possible on this test = 100

Please put your name and email address on the test in two places.

The answers for this midterm <== Incomplete expressionrm are posted here:

https://sites.google.com/a/fullerton.edu/activeprofessor/5-past-semesters/2021-fall/cpsc-240-01

Everyone is encouraged to compare the posted answer key with his or her own test answers. If a grading error is uncovered be sure to contact the instructor to recover any lost points.