LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

**Quiz No.2 PLEASE SUBMIT ON SLACK by 1:40 PM**

October 25 , 2021

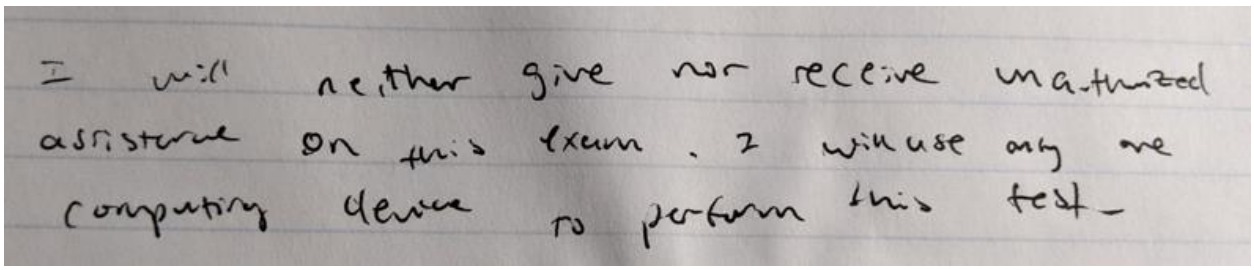
Please circle around your major: **Computer Science** or **Computer Engineering**

**NO CORRECTIONS ARE ALLOWED on FRONT page!!!!!**

You may use the back page for computations. Please answer all questions. **Not all questions are of equal difficulty. Please review the entire quiz first and then budget your time carefully.**

*MAX NUMBER OF POINTS YOU CAN GET IN THIS TEST IS 100.*

***SIGN:***


I will neither give nor receive unauthorized assistance on this exam. I will use only one computing device to perform this test.

***NOTE: Answers given without justification - NO CREDIT FOR THE QUESTION!!!!!***

**Question 1. ( 30 Points)** **Memory model is a linear array of bytes, as shown in Figure 1. The minimal addressable unit in this memory is one byte. Below, Figure 1. depicts a small part of such a memory. The absolute address 0x80000100 is used as a base address and is stored in a register RBase . For clarity, this address is depicted to the left of the corresponding byte.**

30/30

LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

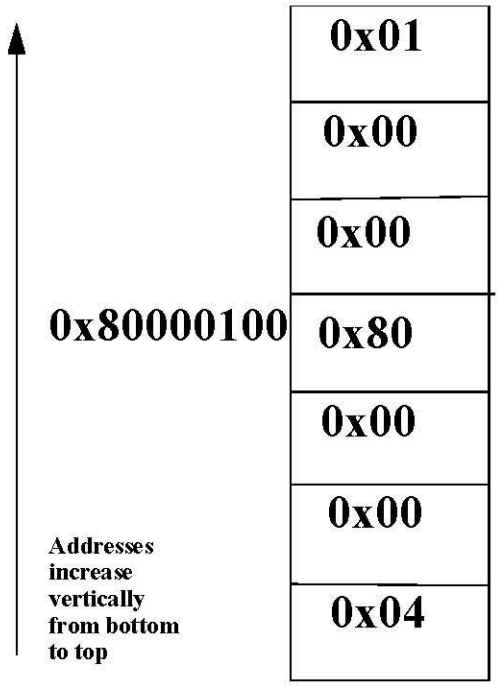



**Figure 1.** Memory model is a linear array of bytes.

Q.1.1. [5 points] Assume you have a MIPS processor and associated memory, as shown in Figure 1. What is the signed decimal value of the 32 bit integer (word) at the address0x80000100?

-2^31 + 1 is the signed dec value, mips is big endian and 0x80 is most sig and 0x01 is least sig.
1 0000000 0000 0000 0000 0000 0000 0001 = -2^31 + 1

Q.1.2. [5 points] Assume you have an INTEL i7 processor and associated memory, as shown in Figure 1. What is the signed decimal value of the 32 bit integer (word) at the ad-dress 0x80000100?

2^24 + 2^7 because intel is little endian so most sig bit is -2^31 and 0x01 is least sig.
00000001 00000000 00000000 10000000 = 2^24 + 2^7

Q.1.3. [5 points] what is the address of a byte containing 0x01?

Based on figure 1, 0x80000103, 3 address values from base pointer

Q.1.4 [5 points] what is the offset from base address (stored in Register RBase) to the byte containing 0x01?

+3, 3 addr up from base pointer

Q.1.5. [5 points] what is the address of a byte containing 0x04?

LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

08x800000FD, 3 addr from bottom

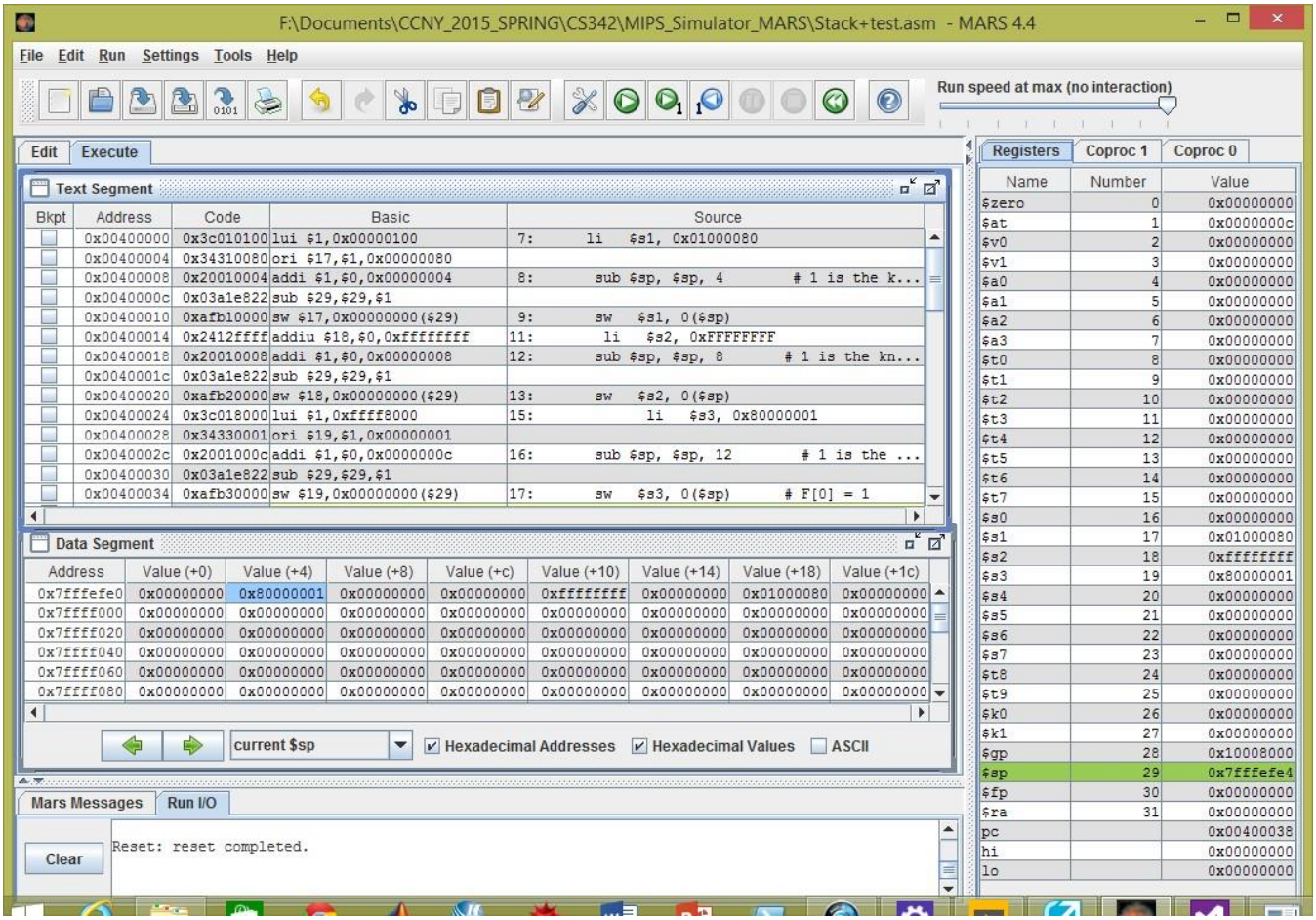
Q1.6 [5 points] what is the offset from base address (stored in Register RBase) to the byte containing 0x04?

0x80000100 – 0x800000FD = -3

25/25

## Question 2. ( 25 Points)

You are given an instance of a program in MARS MIPS simulator window.




2.1. [5 points]What is the signed decimal value of the integer on top of the stack.?

$sp = 0x7fffefe4 = 0x7fffefe0 + 4 = 0x80000001 = -2^31+1

2.2 [5 points]What is the value stored in stack pointer register?

LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

$sp = stack pointer = 0x7fffefe4

2.3.1 [2.5 points]Compute the address of an integer stored on the stack at offset **+12** from the stack pointer.

0xe4 + 0x0c = F0 = 1111 0000

2.3.2 [2.5 points]What is the signed decimal value of the integer at this location?

-1, 2's complement 0xffffffff = 000…01 = -1

2.4.1 [2.5 points]Compute the address of an integer stored on the stack at offset **+20** from the stack pointer.

0xe4 + 0x14 = 0xf8 = 1111 1000

2.4.2 [2.5 points]What is the signed decimal value of the integer at this location?

2^7 + 16^6 + 8 x 16 = 0x10000080

2.5[5 points] Can you determine the address of the instruction that will be executed next step? If yes, please write it down.

0x00400038, look at pc reg

## Question 3. (35 points)

25/35

**You are using MS Visual Studio development environment. The processor is Intel i7.**
**In DEBUG mode you display REGISTER, DISASSEMBLY, and MEMORY windows.**
**Please answer the following questions based on the information displayed in the DEBUG mode windows.**

1. *(1 points)* What is the content (what number is stored in EBP) of the base pointer register EBP?

   0x006CF9E4, look in register EBP

2. *(1 points)*Can you specify the Memory window # where partial ***Stack Frame*** is displayed? If YES , please YES and give the window #. If **No,** Please write NO.

   Yes, Window #3

3. *(10 points)*Based on the information shown in the screenshots, can you determine if variable ***m*** is static or local? Please circle around your choice word. If it is possible, to answer questions

LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

Local, in disassembly window it is in the stack

3.1. What is the offset from base pointer to local variable ***m*** on the stack?

0xd4 = 1101 0100 = -128 + 88 = -44

3.2. Please list all absolute addresses to the **offsets of** variable ***m*** as used in instructions the program:

0x00EB13D5 = 0x00eb13d3 + 0x02 = 0x00eb13d5
0x00EB13DF = 0x00eb13dd + 0x02 = 0x00eb13df

3.3. What is the address of local variable ***m*** on stack?

0xe4 + 0xd4 = 1011 1000 = 0xb8

3.4. What is the signed value (in DECIMAL) of local variable ***m*** as you can observe on ***Stack Frame***?

Int m = EFFF FFFF in disassembly. EFFF FFFF = -2 in 2's complement

4. *(10 points)*Based on the information shown in the screenshots, can you determine if variable ***quizint*** is static or local? Please circle around your choice word.

Local, it is in the stack

4.1. What is the offset from base pointer to local variable ***quizint*** on the stack?

0xf8 = 1111 1000 = -8 in 2's complement

4.2. Please list all absolute addresses to the offsets of variable ***quizint*** as used in the program:

0x00eb13be + 0x02 = 0x00eb13c0

4.3. What is the address of local variable ***quizint*** on stack?

0xe4 + 0xf8 = dc, offset + EBP

# CSC 342

4.4. What is the signed value (in DECIMAL) of local variable ***quizint*** as you can observe in ***Stack Frame***?

0x006cf9dc = 0x01000050 = 2^24 + 2^7

5. *(10 points)*Based on the information shown in the screenshots, can you determine if variable ***MIPSInt*** is static or local? Please circle around your choice word.

Local variable, it in stack

5.1. What is the offset from base pointer to local variable ***MIPSInt*** on the stack?

0xe0 = 1110 0000 = -32

5.2. Please list all absolute addresses to the offsets of variable ***MIPSInt*** as used in the program:

0x00eb13ce = 0x00eb13cc + 0x02

5.3. What is the address of local variable ***MIPSInt*** on stack?

0xe4 + 0xe0 = c4 = 1100 1100 → 0x006cf9c4

5.4. What is the signed value (in DECIMAL) of local variable ***MIPSInt*** as you can observe in ***Stack Frame***?

Addr 0x006f9c4 = 0x80000001 = -2^31 + 1

6. *(1 points)*Can you determine the address of the instruction that will be executed next instance?

0x00eb13dd, look at EIP

7. *(1 points)*What is the assembly code length in bytes?

Last IP – first IP + 1 = EB – A0 + 1 = 1000 1011 = 76

8. *(1 points)*Can you determine the number of instruction of length 7 bytes? If yes, What is it?

00eb13be, 00eb13c5, 00eb13cc, 00eb13d3.. look at disassembly window

LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

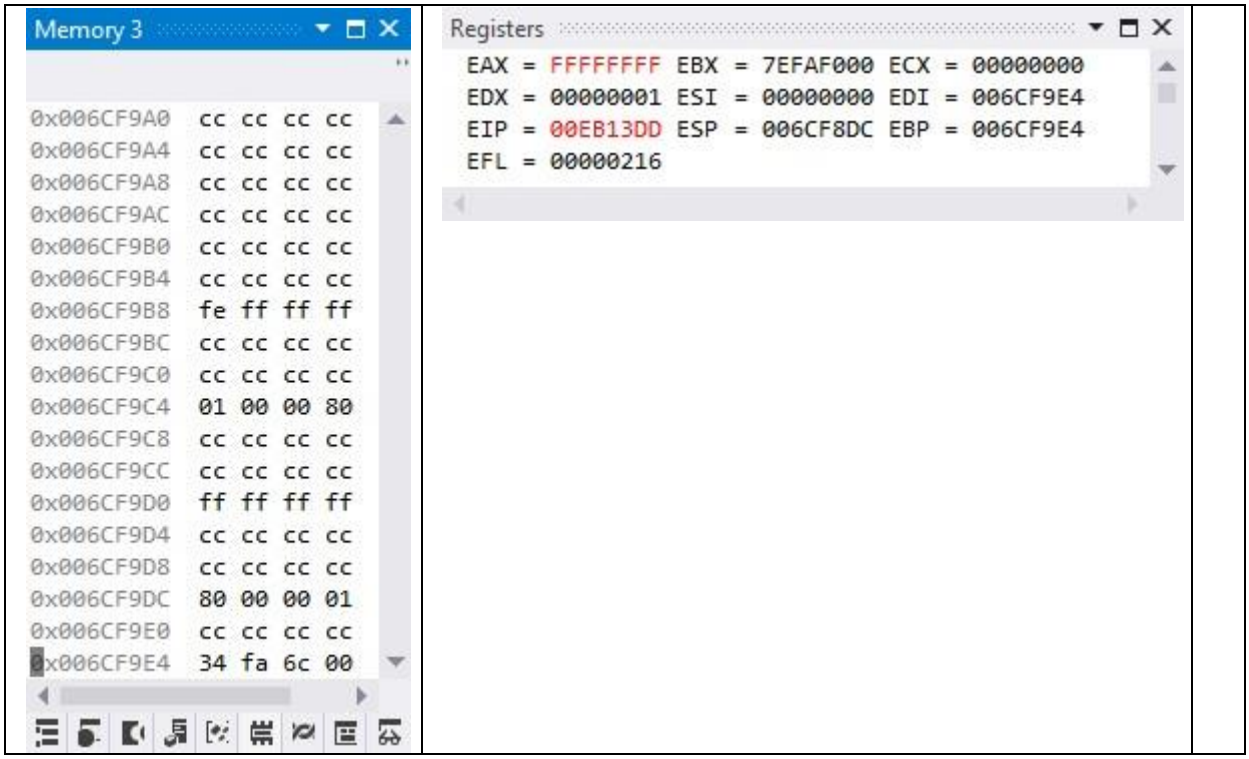
9. *(1 points)*Can you determine the number of instruction of length 6 bytes? If yes, What is it?

00eb13a3, 00eb13ac, look at disassembly window

10. *(1 points)*Can you determine the number of instruction of length 5 bytes? If yes, What is it?

00eb13b2 and 00eb137, look at disassembly window

**Question 3. ( cont'd )**



LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

**Question 3. ( cont'd )**

Disassembly

Address: main(void)

Viewing Options

```
     1: void main()
     2:  {
00EB13A0 55                    push        ebp
00EB13A1 8B EC                 mov         ebp,esp
00EB13A3 81 EC FC 00 00 00     sub         esp,0FCh
00EB13A9 53                    push        ebx
00EB13AA 56                    push        esi
00EB13AB 57                    push        edi
00EB13AC 8D BD 04 FF FF FF     lea         edi,[ebp-0FCh]
00EB13B2 B9 3F 00 00 00        mov         ecx,3Fh
00EB13B7 B8 CC CC CC CC        mov         eax,0CCCCCCCCh
00EB13BC F3 AB                 rep stos    dword ptr es:[edi]
     3:      int quizint = 0x01000080;;
00EB13BE C7 45 F8 80 00 00 01 mov          dword ptr [quizint],1000080h
     4:      int n = 0xFFFFFFFF;
00EB13C5 C7 45 EC FF FF FF FF mov          dword ptr [n],0FFFFFFFFh
     5:      int MIPSInt = 0x80000001;
00EB13CC C7 45 E0 01 00 00 80 mov          dword ptr [MIPSInt],80000001h
     6:      int m =-2;
00EB13D3 C7 45 D4 FE FF FF FF mov          dword ptr [m],0FFFFFFFEh
     7:      int f;
     8:      f=(n-m);
00EB13DA 8B 45 EC              mov         eax,dword ptr [n]
00EB13DD 2B 45 D4              sub         eax,dword ptr [m]
00EB13E0 89 45 C8              mov         dword ptr [f],eax
     9:  }
00EB13E3 33 C0                 xor         eax,eax
00EB13E5 5F                    pop         edi
00EB13E6 5E                    pop         esi
00EB13E7 5B                    pop         ebx
00EB13E8 8B E5                 mov         esp,ebp
00EB13EA 5D                    pop         ebp
00EB13EB C3                    ret
```

LAST NAME: ZHANG FIRST NAME : CHUE

# CSC 342

***In EACH Questions 4.1-4.2 you are given SIGNED Integers stored in 16 BIT Registers. If there is an overflow, please indicate.***

5/5

**4.1 [5 points]** What is the result (hexadecimal, decimal and binary) of the following subtraction:

```
  0x7FFF
 -
0xFFFF
```

HEX: -8000
Decimal: 32767 – -1 = 32768
Binary: 0111111111111111 – 1111111111111111 = 1000 0000 0000 0000

overflow [-32768, 32767]

5/5

**4.2 [5 points]** What is the result(hexadecimal, decimal and binary) of the following addition:

```
  0x7FFF
 +
0xFFFF
```

HEX: 17ffe
Decimal: 32767 + -1 = 32766
Binary:
0111111111111111 + 1111111111111111 = 0111 1111 1111 1110

No Overflow, [-32768, 32767]