Quiz-3-Repeat CS321-8thNovember2020

Total points 36/50

Email address *

maheeth2000@gmail.com

✓ Assembler turns the assembly language code into an object file containing machine language code. Assuming two pass assembly process. In First pass 1/1

- ( ) assembler assigns and instruction addresses
- ( ) finds all the symbols
- ( ) Identifies labels and global variable names
- (•) All of the above ✓

✓ In MIPS, stack pointer register is (give answer in decimal) 1/1

29 ✓

✓ Assembler turns the assembly language code into an object file containing machine language code. Assuming two pass assembly process. In second pass 1/1

- ( ) assembler assigns and instruction addresses
- ( ) finds all the symbols
- ( ) Identifies labels and global variable names
- (•) produces the machine language code ✓
- ( ) All of the above

✗ Which of the following statements is correct? 0/2

- ( ) A word is always two bytes
- ( ) A Unicode character can be stored in 8 bits
- ( ) On a little endian processor, the least significant byte of a word can be found at the same address as the word
- (•) All are correct ✗

Roll Number *

1801CS31

✓ Consider the following program fragment in MIPS32 assembly code:Which of the following statements are correct? 2/2

```
li $s0, -1
srl $v0, $s0, 1
addiu $a0, $v0, 1
```

- ○ Register $a0 will contain the largest positive representable signed number after executing the fragment
- ○ Register $v0 will contain −1 after executing the fragment
- ◉ The fragment will occupy 128 bits in memory ✓
- ○ All of the above

✓ The MIPS instruction set includes several shift instructions. They include logical shift left, logical shift right, and arithmetic shift right. Why doesn't MIPS offer an "arithmetic shift left" instruction? 1/1

- ○ Not essential
- ○ Difficult to implement
- ◉ there's no sign bit on the right-hand side of a register to extend ✓
- ○ None of the above

✓ Suppose that " MemWrite" in the single-cycle MIPS processor has a stuck-at-0 fault, meaning that the signal is always 0,regardless of its intended value. What instructions would malfunction? 1/1

- ○ R-type
- ◉ sw ✓
- ○ addi
- ○ all of the above

✓ Suppose that " ALUOp1" in the single-cycleMIPS processor has a stuck-at-0 fault, meaning that the signal is always 0,regardless of its intended value. What instructions would malfunction? 1/1

- ◉ R-type ✓
- ○ lw
- ○ addi
- ○ all of the above

✓ Consider the following program fragment in MIPS32 assembly code:Which of the following statements are correct? 2/2

```
li $s0, -1
srl $v0, $s0, 1
addiu $a0, $v0, 1
```

- ○ Register $a0 will contain the largest positive representable signed number after executing the fragment
- ○ Register $v0 will contain −1 after executing the fragment
- ○ The fragment will occupy 96 bits in memory
- ◉ Register $a0 will contain the least negative representable signed number after executing the fragment ✓

✓ What are the typical features of the von Neumann architecture? 1/1

- ◉ The processor has a special register called the program counter (PC). ✓
- ○ Each machine instruction contains the address of the next instruction
- ○ Programs have to be written in assembly language
- ○ All of the above
- ○ Other: .....

✓ Select from the following examples which meets architecture design principles: simplicity favors regularity 1/1

- ○ Each instruction has a 6-bit opcode
- ○ MIPS has only 3 instruction formats (R-Type, I-Type, J-Type).
- ○ Each instruction format has the same number and order of operands
- ○ Each instruction is the same size, making decoding hardware simple
- ◉ All of the above ✓

✓ Suppose that "RegWrite" in the single-cycleMIPS processor has a stuck-at-0 fault, meaning that the signal is always 0,regardless of its intended value. What instructions would malfunction? 1/1

- ○ R-type
- ○ lw
- ○ addi
- ◉ all of the above ✓

✓ bne instruction has x bits for address (give answer in decimal) 2/2

16 ✓

✗ The value of c is ···/3

Consider the following code, being executed on a Little Endian MIPS machine where

- sizeof(int) == 4;sizeof(int *) == 4; sizeof(char) == 1

For each of the following assignment statements, fill in the blanks in the comments to indicate the result of the assignment. All answers must be in hex.

```
int main() {
int array[2];
int  *ptr;
int x; char c;
array[0] = 0xaabbccdd;
array[1] = 0x44556677;
ptr = array;
x = *((int *)ptr + 1);
c = *((char *)ptr + 1);
x = *((char *)ptr + 1);
c = *((int *)ptr + 1);
/* c = 0x_____*/}
```

0x66 ✗

✗ In MIPS procedure call , Save any registers that are needed ($s0-$s7) is the responsibility of the ..... ···/1

user ✗

Name *

Maheeth Reddy

✗ The value of x is 0/3

Consider the following code, being executed on a Little Endian MIPS machine where

- `sizeof(int) == 4;sizeof(int *) == 4; sizeof(char) == 1`

For each of the following assignment statements, fill in the blanks in the comments to indicate the result of the assignment. All answers must be in hex.

```
int main() {
int array[2];
int  *ptr;
int x; char c;
array[0] = 0xaabbccdd;
array[1] = 0x44556677;
ptr = array;
x = *((int *)ptr + 1);
/* x = 0x_____*/
```

- 0x44556677
- 0xffffff77 ✗
- 0xffffffcc
- 0xaabbccdd

✓ Assuming the content as shown. The value of $3 after executing lw $3, 12($1) 2/2

Register	Value
$1	12
$2	16
$3	20
$4	24

Memory Address	Value
12	16
16	20
20	24
24	28

28 ✓

✓ To implement Pseudoinstruction li $s0, 0x1234AA77 X MIPS Instructions required 2/2

2 ✓

✗ Assuming the content as shown. The value of $3 after executing sll $3, $3,2 …/1

Register	Value
$1	12
$2	16
$3	20
$4	24

Memory Address	Value
12	16
16	20
20	24
24	28

5 ✗

✓ Opcode for floating points instructions in MIPS is (give answer in decimal) 2/2

17 ✓

✓ Assuming the content as shown. The value of $3 after executing add $3, $2, $1 2/2

Register	Value
$1	12
$2	16
$3	20
$4	24

Memory Address	Value
12	16
16	20
20	24
24	28

28 ✓

✓ Assuming the content as shown. The value of $2 after executing addi $2, $3, 16 2/2

Register	Value
$1	12
$2	16
$3	20
$4	24

Memory Address	Value
12	16
16	20
20	24
24	28

36 ✓

✗ In MIPS procedure call , Save any registers that are needed ($ra, maybe $t0-t9) is the responsibility of the ..... ···/1

user ✗

✓ Assuming the content as shown. The value of $3 after executing addi $2, $3, 16 1/1

Register	Value
$1	12
$2	16
$3	20
$4	24

Memory Address	Value
12	16
16	20
20	24
24	28

20 ✓

✓ The value of x is 3/3

Consider the following code, being executed on a Little Endian MIPS machine where

- `sizeof(int) == 4;sizeof(int *) == 4; sizeof(char) == 1`

For each of the following assignment statements, fill in the blanks in the comments to indicate the result of the assignment. All answers must be in hex.

```
int main() {
int array[2];
int  *ptr;
int x; char c;
array[0] = 0xaabbccdd;
array[1] = 0x44556677;
ptr = array;
x = *((int *)ptr + 1);
c = *((char *)ptr + 1);
x = *((char *)ptr + 1);
/* x = 0x_____*/
```

0xffffffcc ✓

✓ text segment stores the machine language program. It is large enough to accommodate almost X MB of code 2/2

- ○ 1024
- ○ 128
- ○ 512
- ◉ 256 ✓
- ○ Other: .....

✓ In MIPS, Jump Register equivalent to ..... instruction X86 1/1

RET ✓

✓ Which of the following statements best describes the use of the rt field by the MIPs sw instruction 1/1

- ○ The contents of the rt register are added to the signed 16-bit constant and used as a memory address
- ○ The contents of the rt register are added to 4 times the signed 16-bit immediate field and used as a memory address
- ◉ The contents of rt are stored into memory ✓
- ○ The contents of the memory location with the address pointed to by rt are stored in memory
- ○ The 5-bit value encoded in the rt field is stored in memory

✗ The value of c is 0/3

Consider the following code, being executed on a Little Endian MIPS machine where

- sizeof(int) == 4;sizeof(int *) == 4; sizeof(char) == 1

For each of the following assignment statements, fill in the blanks in the comments to indicate the result of the assignment. All answers must be in hex.

```
int main() {
int array[2];
int  *ptr;
int x; char c;
array[0] = 0xaabbccdd;
array[1] = 0x44556677;
ptr = array;
x = *((int *)ptr + 1);
c = *((char *)ptr + 1);
/* c = 0x_____*/
```

- oxdd
- ox66
- (●) 0x77 ✗
- 0xcc

✓ Assume that x, y, z, q are stored in registers $s1-$s4. Minimum number of MIPS instructions required to implement expression x = x + y + z - q is 1/1

- 2
- 4
- 5
- (●) 3 ✓

✓ In MIPS, global variables are accessed using the global pointer ($gp), which is initialized to 1/1

- ○ 300080000
- ○ 200080000
- ○ 10000000
- ○ 700080000
- ◉ 100080000 ✓

✓ Select from the following examples which meets architecture design principles: Make the common case fast 1/1

- ○ Registers make the access to most recently accessed variables fast
- ○ The RISC (reduced instruction set computer) architecture, makes the common/ simple instructions fast because the computer must handle only a small number of simple instructions
- ○ Most instructions require all 32 bits of an instruction, so all instructions are 32 bits. The instruction size is chosen to make the common instructions fast.
- ◉ All of the above ✓