

CS 4400 Fall 2018

Midterm Exam 1

Name: TAKU SAKIKAWA

Instructions You have eighty minutes to complete this open-book, open-note exam. Electronic devices are allowed only to consult notes or books from local storage; network use is prohibited. **Write only on the front side of each page**, and ask the proctor for extra pages if needed.

For the next four questions, assume the following register and memory state:

CPU		Memory	
register	value	address	value (8 bytes)
rax	0x0004	0x0440	0x001
rbx	<u>0x0450</u>	0x0448	0x002
rcx	0x0018	0x0450	<u>0x040</u>
rdx	<u>0x0428</u>	0x0458	0x420
rbp	<u>0x7FA8</u>	0x0460	0x440

1. What is the value of $-0x10(\%rbp)$ as a source argument to `leaq`?

6 points

$$0x7FA8 - 10 = 7F98$$

Value = 0x7F98

2. What is the value of $0x10(\%rbx)$ as a source argument to `movq`?

6 points

$$rbx = 0450$$

$$0450 + 10 = 0460 \rightarrow 0x440$$

Value = 0x440

3. What is the value of $(\%rcx, \%rdx)$ as a source argument to `movq`?

6 points

$$rcx = 18$$

$$rdx = 428$$

$$rcx + rdx = 440 \rightarrow 0x001$$

Value = 0x001

4. What is the value of $0x8(\%rdx, \%rax, 8)$ as a source argument to `addq`?

6 points

$$rax = 4$$

$$rdx = 428$$

$$rax \cdot 8 + rdx + 8$$

$$8 \cdot 4 + 428 + 8 = 450 \rightarrow 040$$

Value = 0x040

edi a
rsi b
rdx c

The next three questions refer to compare defined as

```
int compare(TYPE a, TYPE b, TYPE c) {
    return (VAR1 < VAR2);
}
```

where the macros VAR1 and VAR2 can be defined as a, b, or c, and the macro TYPE can be defined as int, long, or unsigned.

5. What definitions of TYPE, VAR1, and VAR2 are consistent with the following compiled form of compare? 6 points

below

```

xorl %eax, %eax
cmpl %edi, %esi
setb %al
ret

```

$eax = 0$
 $esi < edi < 0$
 $v1 < v2$
 $b < a$

TYPE = unsigned VAR1 = b VAR2 = a

6. What definitions of TYPE, VAR1, and VAR2 are consistent with the following compiled form of compare? 6 points

sign

```

xorl %eax, %eax
cmpq %rdx, %rdi
setl %al
ret

```

$a < c$
 $rdi < rdx < 0$
 $v1 < v2$

TYPE = long VAR1 = a VAR2 = c

7. What definitions of TYPE, VAR1, and VAR2 are consistent with the following compiled form of compare? 6 points

int
sign

```

xorl %eax, %eax
cmpl %esi, %edi
setg %al
ret

```

$edi > esi > 0$
 $esi < edi$
 $b < a$

TYPE = int VAR1 = b VAR2 = a

3 = 1

8. After running the assembly sequence

10 points

```

andl $0x3, %eax  —  $eax \& 3 \rightarrow eax \dots 3 \text{ or less}$ 
andl $0x3, %ebx  —  $ebx \dots 3 \text{ or less}$ 
cmpl %ebx, %eax  —  $(eax \leq ebx) \leq 0 \rightarrow L1$ 
jle .L1
subl $0x1, %eax  —  $eax = eax - 1 \text{ (if } eax > ebx)$ 
.L1:
andl %ebx, %eax  —  $ebx \& eax$ 

```

what are all the possible values of register %eax (making no assumptions about the initial values of %eax and %ebx)?

Possible values = 3, 2, 1, 0

To enable partial credit, show what must be true of the value of %eax and %ebx after each instruction of the assembly code above.

$$0 \leq eax \leq 3$$

$$0 \leq ebx \leq 3$$

$$eax - ebx \leq 0$$

$$011$$

$$\nearrow \text{max } 3 = 11$$

$$\text{and } \%ebx \quad \%eax$$

$$\begin{array}{ccc} 3 & 2 & 1 \\ 011 & 010 & 001 \end{array}$$

$$\text{max of } eax = 11$$

$$\text{min of } eax \& ebx = 0$$

$$\begin{array}{c} 0 \\ 000 \end{array}$$

$$0 \leq 0 = 00$$

$$\text{min of } eax = 00$$

3

between $3 \sim 0$
 $11 \sim 00$ which is 0, 1, 2, 3

$di = a$ (int)
 $rsi = c$ (int)
 $rdi = b$ (long)

✓ ① $rdi \dots 4\text{byte} = a$

② $rsi = c$

✓ ③ $rdx \dots 8\text{byte} = b$

~~④~~ $rcx = edi = a$

$rax = rcx = edi = a$ 12 points

$b < a$

9. Given that the function go

```

int go(ARGS) {
    while (COMP)
        a OP c;
    if b > a → return a;
}
  
```

$a \leq c$

compiles as

$b \leq c$

```

movslq %edi, %rcx
cmpq   %rcx, %rdx
movq   %rcx, %rax
jl     L2

L3:
subl   %esi, %eax
movslq %eax, %rcx
cmpq   %rdx, %rcx
jle    L3

L2:
ret
  
```

$rcx = edi$

$rdx < rcx < 0$

$rax = rcx = edi = a$

$b < a$ or $a \geq b$

$a - c$

$eax - esi \Rightarrow a = a - c \dots a -= c$

$rcx = eax \Rightarrow rcx = \text{old } a = \text{new } a$

$rcx - rdx \Rightarrow a - b \leq 0 \quad a \leq b$

$a - b \leq 0$

$a \leq b$

pick a combination of ARGS, COMP, and OP (not necessarily in the same row) that fits, circling one choice in each column:

ARGS	COMP	OP
int a, int b, int c	$a >= b$	$=$
int a, int c, long b	$c >= a$	$+=$
long a, int c, int b	$a < c$	$-=$
int a, unsigned b, int c	$b >= a$	$!=$

To enable partial credit, show your work disassembling the code above, including mapping registers to C variables.

int a, int c, long b

$b >= a$

$-=$

rdi ... i
rsi ... j
rdx ... mat1
rcx ... mat2

10. Given that

```
int sum_element(int i, int j,
                int mat1[][M], int mat2[][N]) {
    return mat1[i][j] + mat2[i][j];
}
```

12 points

compiles as

```
movslq %edi, %rdi    rdi = i
movslq %esi, %rsi    rsi = j
leaq 0(,%rdi,4), %r8  r8 = 4i
movq %rdi, %rax      rax = i
sarlq $5, %rax       rax = 32i
addq %r8, %rdi       rdi = 5i
subq %r8, %rax       rax = 28i
addq %rax, %rdx       rdx = mat1 + 28i
leaq (%rcx,%rdi,8), %rax  rax = mat1 + 40i
movl (%rax,%rsi,4), %eax  eax = mat1 + 40i + 4j
addl (%rdx,%rsi,4), %eax  eax = (mat1 + 40i + 4j) + (mat2 + 28i + 4j)
ret
```

$6i + mat1$
 $4j + mat1 + 8i$
 $4j + mat2 + 28i + 4j$

then what are the values of the constants M and N among the following possibilities?
(Circle one.)

- M = 8 and N = 9
- ☒ M = 7 and N = 10
- M = 6 and N = 11
- M = 5 and N = 12
- M = 4 and N = 13
- M = 3 and N = 14

$mat2 + 28i$
 $i + 4i$
 $4j + rax \quad 32i - 4i = 28i$
 $5i \times 8 + mat1$
 $40i + mat1$

To enable partial credit, show your work disassembling the code above.

$\begin{pmatrix} mat2 & 28i & 4j \end{pmatrix}$
 $\begin{pmatrix} mat1 & 8i & 4j \end{pmatrix}$

40 4

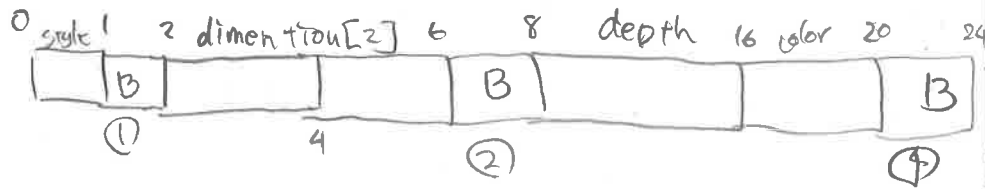
$2 \begin{bmatrix} 10 & 1 \end{bmatrix}$

$10 \times M + 1$

$10M + 1$

$\begin{bmatrix} 7 & 1 \end{bmatrix}$

$7N + 1$



11. Given the declarations

12 points

```
typedef struct present {
    char style;
    short dimensions[2];
    double depth;
    int color;
} present;
```

present p[1000];

and if the array p starts at address 0x1000, then what is the address of p[2].dimensions[1]?

Address = 0x1034

To enable partial credit, show relevant sizeof and/or offset of calculations.

$$2 \times 24 = 48$$

$$2 \times 24 + 4 + 0x1000 = \underline{0x1034}$$

$$1 + (2 + 8) + (4) + 1 + (2 + 4)$$

$$\underline{22}$$

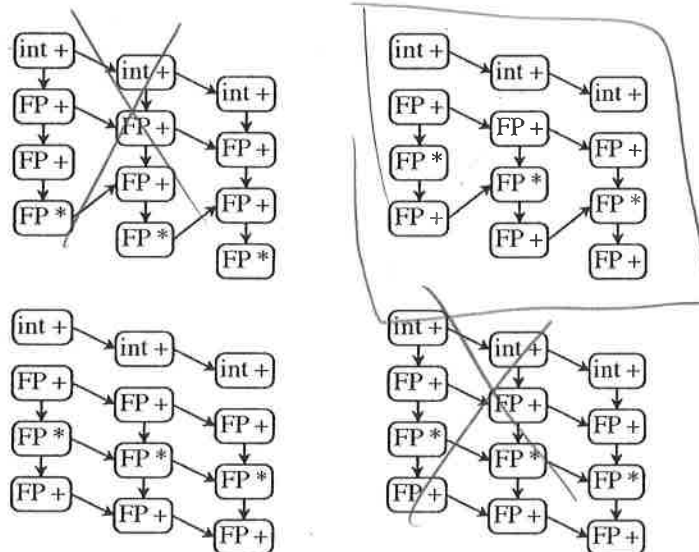
$$\begin{matrix} 10 & 10 & 4 & & 1 & & 4 \\ \textcircled{1} + \textcircled{4} + \textcircled{8} + \textcircled{4} + \textcircled{1} + \textcircled{2} + 4 \\ \underline{22} \end{matrix}$$



Diagram illustrating the evaluation of the expression $(int) \rightarrow FP+ \rightarrow FP* \rightarrow FP+$ using a stack:

- Initial state: Stack contains int .
- Operation $+$: The stack contains $FP+$.
- Operation $*$: The stack contains $FP*$.
- Operation $+$: The stack contains $FP+$.

$10 + +$ ①
 $FP +$ ③
 $FP *$ ⑤



Cycles per steps = $\frac{8}{(3) \cdot (4) \text{ steps}}$

