

어셈블리 Take-home Exam

문항 수: 10

2020년 2학기

0[배점 없음]. 학번 및 이름을 쓰시오.

학번: 이 름:

1[2]. SPARC 프로세서는 다음 중 어느 기계에 속하는가?

- ① : 스택 기계 (stack machine)
- ② : 단일 레지스터 기계 (accumulator machine)
- ③ : 다중 레지스터 기계 (load/store machine)

2[8]. 다음 SPARC에 대한 설명의 참 / 거짓을 선택하시오.

- ① 전방전달(forwarding)을 이용하여도 파이프라인 지연을 막을 수 없는 경우가 있다. (참 / 거짓)
- ② 레지스터의 개수가 늘어나면, 기존에 구현할 수 없었던 프로그램 구현이 가능해지기도 한다. (참 / 거짓)
- ③ 메모리 장치와 프로세서 사이 가능한 데이터 이동단위는 1바이트, 2바이트, 그리고 4바이트이다. (참 / 거짓)
- ④ 레지스터 %l0 값이 13일 경우, 명령어 “btst 0x19, %l0”를 실행하면 condition code 중 Z값이 0으로 세팅된다. (참 / 거짓)

3[20]. 다음 SPARC 어셈블리 표현 중 잘못된 형식 여부를 참 / 거짓으로 표시하고, 거짓인 경우는 잘못된 부분을 설명하시오.

- | | |
|-----------------------|----------|
| ① add 10, %l0, %l1 | (참 / 거짓) |
| ② addcc %l0, %l1, %l2 | (참 / 거짓) |
| ③ ld %l0, [%l1 + 1] | (참 / 거짓) |
| ④ st [%l0 + 0], %o0 | (참 / 거짓) |
| ⑤ cmp %l0, 0x2345 | (참 / 거짓) |
| ⑥ VAL = 10 + 2 * 5 | (참 / 거짓) |

⑦ ld [%i1 - %i2], %i0 (참 / 거짓)

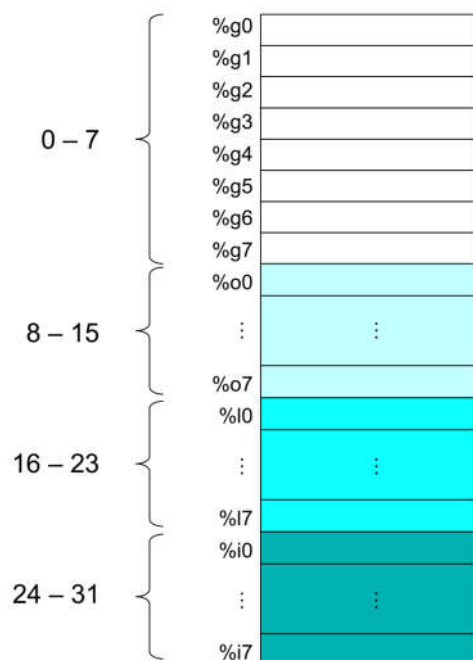
⑧ st %i5, [%i1 - 100] (참 / 거짓)

⑨ ldd [%o1+0x8], %o3 (참 / 거짓)

⑩ or %o2, %e2, %o3 (참 / 거짓)

4[10]. 다음 정보를 참고하시오.

※ SPARC 레지스터 번호



※ 산술 및 논리 연산 명령어 형식

비트번호	31 30	29	25	24	19	18	14	13	12	5	4	0
의미	OP	S		OP-확장	R			0				A

비트번호	31 30	29	25	24	19	18	14	13	12		0
의미	OP	S		OP-확장	R			1		상수 (A)	

OP 는 “10₂”

OP-확장:

명령어	OP-확장
add	000000
and	000001
or	000010
xor	000011
sub	000100
andn	000101
orn	000110
xnor	000111
addx	001000
subx	001100
addcc	010000
andcc	010001
orcc	010010

명령어	OP-확장
xorcc	010011
subcc	010100
andncc	010101
orncc	010110
xnorcc	010111
addxcc	011000
subxcc	011100
sll	100101
srl	100110
sra	100111
jmp1	111000
save	111100
restore	111101

4.1[5]. 다음 SPARC 어셈블리 명령어를 16진수(hex) 기계코드(machine code)로 표현하시오.

```
xnorcc %l1, -2, %l2
```

4.2[5]. 다음 16진수(hex) 기계코드(machine code)를 SPARC 어셈블리 명령어로 표현하시오.

```
0x80A7C00F
```

5[10]. 다음은 %l0 레지스터 값, %l1 레지스터의 값, 그리고 실행할 명령어이다. 명령어 실행 후, condition code Z, N, V, C 및 %l2 레지스터 값을 표에 적으시오.

```
5.1[5]. %l0: 0x8000_0000
        %l1: 0x8000_0000

        addcc %l0, %l1, %l2
```

Z	N	V	C	%l2

```
5.2[5]. %l0: 0x0123_4567
        %l1: 0xFEDC_BA98
```

```
xorcc %l0, %l1, %l2
```

Z	N	V	C	%l2

6[10]. 다음 코드의 지연주기를 최적화하시오.

※ 힌트: 의사명령어 set을 두 개의 기계명령어로 분리한 후, 최적화하시오.

```
.text
.global main

main:
    save    %sp, -96, %sp
    set     str, %o0
    call    printf
    nop
    ret
    restore

.data
str: .asciz "Hello!\n"
.align 4
```

!! 답안을 여기에 적으시오.

7[10]. 두 개의 양의 정수를 **십진수 형태로** 입력받아 두 수 중에서 작은 수부터 1씩 증가하여 큰 수까지 더하는 어셈블리 프로그램을 작성하고, **코드(mid_prob7.s)를 submit**하시오. (명령: **submit konwoo mid_07**)

※ 가정: 사용자 입력은 32비트 부호 있는 정수 표현들 중 임의의 **양수**이다.

※ 동작 예시:

```
bash $ ./mid_prob7
Value?> 7
Value?> 3
Sum is 25
```

```
bash $ ./mid_prob7
Value?> 3
Value?> 7
Sum is 25
```

```
bash $ ./mid_prob7
Value?> 11
Value?> 20
Sum is 155
```

!! 답안을 여기에 적으시오. 공간이 부족할 경우, 2열(2-column)로 작성 또는 추가페이지 사용 가능.

8[10]. (구현) 네 개의 32비트 정수(A, B, C, D)를 **16진수 형태로** 입력받아 $A*B + C*D$ 를 계산하는 프로그램을 작성하고, **코드(mid_prob8.s)를 submit**하시오. (명령: **submit konwoo mid_08**)

※ 가정: 사용자 입력은 32비트 0 또는 양수이다. 결과는 64비트로 표현.

※ 동작 예시:

```
bash $ ./mid_prob8
Hexadecimal value?> ffffffff
Hexadecimal value?> 8
Hexadecimal value?> ffffffff
Hexadecimal value?> 8
Result is 0000000f fffffff0
```

```
bash $ ./mid_prob8
Hexadecimal value?> ffffffff
Hexadecimal value?> ffffffff
Hexadecimal value?> ffffffff
Hexadecimal value?> ffffffff
Result is fffffffc 00000002
```

!! 답안을 여기에 적으시오. 공간이 부족할 경우, 2열(2-column)로 작성 또는 추가페이지 사용 가능.

9[10]. (구현) 한 개의 양의 정수를 십진수 형태로 입력받아 3의 배수일 경우 “mod 3 = 0”, 3의 배수 더하기 1일 경우 “mod 3 = 1”, 3의 배수 더하기 2일 경우 “mod 3 = 2” 메시지를 출력하는 프로그램을 작성하시오. 또한, **작성한 코드(mid_prob9.s)를 submit**하시오. (명령: **submit konwoo mid_09**)

※ 제한사항: 조건 검사와 내용을 분리하는 switch문 형식으로 구현. bne 사용하지 말 것.

```
bash $ ./mid_prob9
Value?> 360
mod 3 = 0
```

```
bash $ ./mid_prob9
Value?> 211
mod 3 = 1
```

```
bash $ ./mid_prob9
Value?> 3333335
mod 3 = 2
```

!! 답안을 여기에 적으시오. 공간이 부족할 경우, 2열(2-column)로 작성 또는 추가페이지 사용 가능.

10[10]. (구현) 9번과 동일한 문제를 if-else 형식으로 작성하시오. 또한, **작성한 코드(mid_prob10.s)**를 submit 하시오. (명령: **submit konwoo mid_10**)

※ 제한사항: if-else를 통한 구현. be 사용하지 말 것.

```
bash $ ./mid_prob10
Value?> 360
mod 3 = 0
```

```
bash $ ./mid_prob10
Value?> 211
mod 3 = 1
```

```
bash $ ./mid_prob10
Value?> 3333335
mod 3 = 2
```

!! 답안을 여기에 적으시오. 공간이 부족할 경우, 2열(2-column)로 작성 또는 추가페이지 사용 가능.
!! printf가 3개의 인자를 필요로 할 경우, %o0, %o1, %o2를 활용.