Istanbul Technical University – Computer Engineering Department

BLG351E
Quiz Exam

10.01.2022

- Closed books and closed notes; 20 multiple choice questions; 60 minutes.
- Cheating will be penalized according to university disciplinary policy.
- Please fill the form in this page. The answers in the inner pages will not be taken into account.

Student Name:

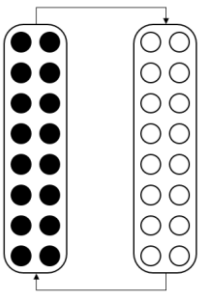
Student ID:

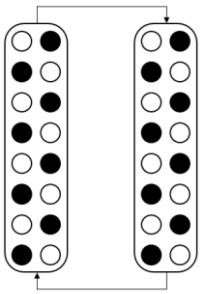
1	(A) (B) (C) (D) (E)
2	(A) (B) (C) (D) (E)
3	(A) (B) (C) (D) (E)
4	(A) (B) (C) (D) (E)
5	(A) (B) (C) (D) (E)
6	(A) (B) (C) (D) (E)
7	(A) (B) (C) (D) (E)
8	(A) (B) (C) (D) (E)
9	(A) (B) (C) (D) (E)
10	(A) (B) (C) (D) (E)
11	(A) (B) (C) (D) (E)
12	(A) (B) (C) (D) (E)
13	(A) (B) (C) (D) (E)
14	(A) (B) (C) (D) (E)
15	(A) (B) (C) (D) (E)
16	(A) (B) (C) (D) (E)
17	(A) (B) (C) (D) (E)
18	(A) (B) (C) (D) (E)
19	(A) (B) (C) (D) (E)
20	(A) (B) (C) (D) (E)

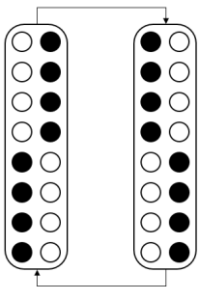
Please answer the following questions according to the given code.

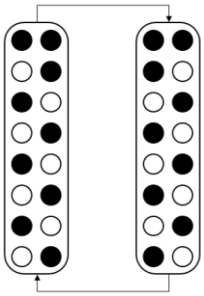
```
 1 SetupP1     bis.b   #11111111b,&P1DIR
 2             bis.b   #11111111b,&P2DIR
 3             mov.b   #10101010b, &P1OUT
 4             mov.b   #01010101b, &P2OUT
 5 Mainloop    xor.b   #11111111b,&P1OUT
 6             xor.b   #11111111b,&P2OUT
 7 Wait        mov.w   #00500000, R15
 8 L1          dec.w   R15
 9             jnz     L1
10             jmp     Mainloop
```

1) What is the pattern obtained using the code above? (White represents the LED is on.)

A) 

B) 

C) 

D) 

E) None of the above.

2) Instead of Lines 5-6, placing which of the following code blocks produces the same pattern?

A)
```
5 Mainloop    rlc.b   &P1OUT
6             rlc.b   &P2OUT
```

B)
```
5 Mainloop    sub.b   #11111111b,&P1OUT
6             sub.b   #11111111b,&P2OUT
```

C)
```
5 Mainloop    mov.b   &P2OUT, R5
6             mov.b   &P1OUT, R6
7             mov.b   R5,&P1OUT
8             mov.b   R6,&P2OUT
```

D)
```
5 Mainloop    mov.b   &P2OUT, R3
6             mov.b   &P1OUT, R6
7             mov.b   R3,&P1OUT
8             mov.b   R6,&P2OUT
```

E)
```
5 Mainloop    mov.b   &P2OUT,&P1OUT
6             mov.b   &P1OUT,&P2OUT
```

3) Instead of Line 5, which of the following lines may be used?

A) `bic.b #11111111b,&P1DIR`

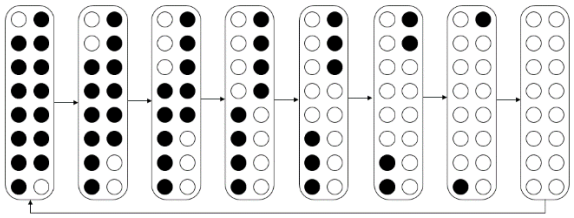
B) `mov.b #11111111b,&P1DIR`

C) `clr.b &P1DIR`

D) `tst.b &P1DIR`

E) `bit.b #11111111b, &P1DIR`

For the LED pattern given below, and the assembly code, fill the blank parts from the code.



```
 1 Setup       mov.b   #0, R5
 2             bis.b   #11111111b,&P1DIR
 3             bis.b   #11111111b,&P2DIR
 4             mov.b   #00000001b, &P1OUT
 5             mov.b   #10000000b, &P2OUT
 6             (X)
 7 Mainloop    mov.b   &P1OUT, R6
 8             (Y)
 9             bis.b   R6,&P1OUT
10             (Z)
11             call #Wait
12             inc     R5
13             (T)
14             jeq     Setup
15             jmp     Mainloop
16
17
18 Wait        mov.w   #01500000, R15
19 L2          dec.w   R15
20             jnz     L2
21             ret
```

4) Find the correct line for (X).

A) `call #Wait` B) `nop`

C) `inc.w R5` D) `inc.b R5`

E) None of the above

5) Find the correct line for (Y).

A) `add.b #1d, &P1OUT` B) `rra.b &P1OUT`

C) `rlc.b &P1OUT` D) `rra.w &P1OUT`

E) None of the above

6) Find the correct line for (Z).

A) `rlc.w &P2OUT` B) `rlc.b &P2OUT`

C) `rra.b &P2OUT` D) `rra.w &P2OUT`

E) `add.b #1d, &P2OUT`

7) Find the correct line for (T).

A) `cmp #00000001b, R5` B) `cmp #10000000b, R5`

C) `cmp #8, R5` D) `cmp #0d, R5`

E) `cmp R6, R5`

Please answer the following questions according to the given code.

```
1               bis.b   #11111111b,&P2DIR
2               mov.b   #00000000b,&P1DIR
3               bic.b   #11111111b,&P2OUT
4               mov.b   #00000100b,&P2OUT
5
6
7 W1            bit.b   #00100000b,&P1IN
8               jz W1
9               bit.b   #00000100b,&P2OUT
10              jnz W2
11 W4           bis.b   #00000100b,&P2OUT
12              bic.b   #00001000b,&P2OUT
13              jmp W3
14 W2           bic.b   #00000100b,&P2OUT
15              bis.b   #00001000b,&P2OUT
16 W3           bit.b   #00100000b,&P1IN
17              jnz W3
18 Wait         mov.w   #00500000, R15
19 W5           dec.w   R15
20              jnz     W5
21              jmp W1
```

8) In which line, a button press is checked?
A) W1 B) W2 C) W3 D) W4 E) W5

9) In which line, a button release is checked?
A) W1 B) W2 C) W3 D) W4 E) W5

10) Instead of Line 2, which of the following assembly line could be used to obtain the same output?

A) `mov.b #11110000b,&P1DIR`

B) `mov.b #11011111b,&P1DIR`

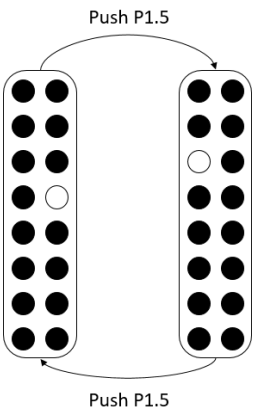
C) `bis.b #11110000b,&P1DIR`

D) `bis.b #11111111b,&P1DIR`

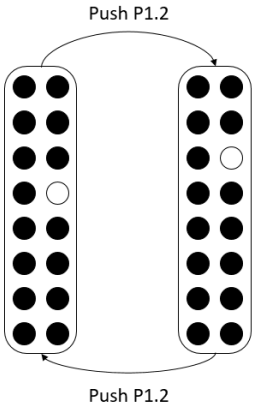
E) `bis.b #00000000b,&P1DIR`

11) What is the pattern obtained by the given code part?

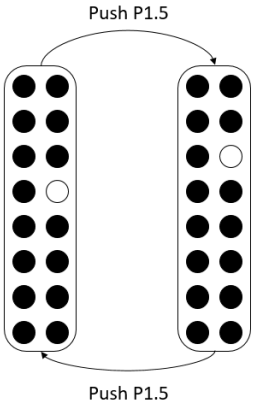
A)



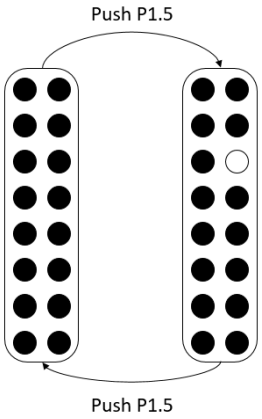
B)



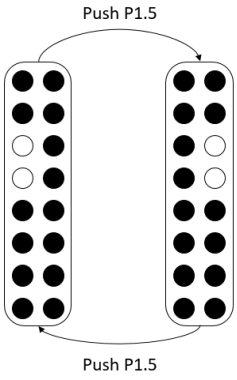
C)



D)



E)



For modulus operation we will perform some steps from Russian Peasant Division which is very useful for binary systems. RPD consists of the following steps for dividend A and divisor B.

- **Create variables C and D, initialize them with B and A's values respectively.**
- **While C is not greater than A/2, multiply it by 2.**
- **While B is not greater than D:**
  - **Subtract C from D wherever D is greater than or equal to C.**
  - **Divide C by 2.**
- **The final value in D gives you the remainder.**

An example implementation is given below.

A	B	C	D
151	8	8	151
151	8	**16**	151
151	8	**32**	151
151	8	**64**	151
151	8	**128**	151
151	8	**64**	**23**
151	8	**32**	23
151	8	**16**	23
151	8	**8**	**7**

```
1            mov.w #arr, R10
2
3            mov.b #0, R11;
4            mov.w #2, R5
5 Label1     mov R5, R6
6 Label2     sub #1, R6
7            cmp #1, R6
8            jeq Label7
9            mov.w R5, R7
10           rra R7
11           mov.w R6,R8
12           mov.w R5,R9
13
14 Label3    cmp R8, R7
15           jl  Label4
16           rla R8
17           jmp Label3
18 Label4    cmp R6,R9
19           jl  Label6
20           cmp R8,R9
21           jl  Label5
22           sub.w   R8,R9
23 Label5    rra R8
24           jmp Label4
25
26 Label6    cmp #0d, R9
27           jne Label2
28           add #1, R5
29           jmp Label1
30
31 Label7    mov.w   R5, 0(R10)
32           add #2d,R10
33           add #1, R5
34           add #1, R11
35           cmp #50, R11
36           jeq fin
37           jmp Label1
38
39 fin       jmp fin
40
41
42           .data
43 arr     .space 100
```

An example assembly code using RPD Modulus is given above. The main purpose of the code is filling an array with Label "arr" according to some pattern.

12) Which variable from the algorithm is kept in R5?
A) A B) A/2 C) B D) B/2 E) C

13) Which variable from the algorithm is kept in R7?
A) A B) A/2 C) B D) B/2 E) C

14) Which variable from the algorithm is kept in R8?
A) A B) A/2 C) B D) C E) D

15) Which variable from the algorithm is kept in R9?
A) A B) B C) B/2 D) C E) D

16) What is the purpose of the algorithm?
A) Fill the array with even numbers.
B) Fill the array with numbers divisible with 50.
C) Fill the array with prime numbers.
D) Fill the array with numbers divisible by 100.
E) Fill the array with odd numbers.

Please answer the following questions according to the given code.

```
1 Setup   mov #array, R5
2         mov #result, R10
3
4 Main    mov.b   @R5, R6
5         inc R5
6         call #func1
7         mov.b   R6, 0(R10)
8         inc R10
9         cmp #lastelement, R5
10        jlo Main
11        jmp finish
12
13
14 func1  xor.b #0FFh, R6
15        mov.b R6, R7
16        call #func2
17        mov.b R7, R6
18        ret
19
20 func2  inc.b R7
21        ret
22
23 finish jmp finish
24
25        .data
26 result .byte   0, 0, 0, 0, 0
27 array  .byte   65, -120, 0, 55, -1
28 lastelement
```

17) Assuming that the initial value of Stack Pointer (SP) is 0x0400, what is the lowest value of the SP along the code?

A) 0x03FE
B) 0x03FC
C) 0x03FA
D) 0x03F9
E) None of the above.

18) What are the final values of "result" array?
A) 66, -119, 1, 56, 0
B) 64, -121, -1, 54, -2
C) -65, 120, 0, -55, 1
D) -66, 119, -1, -56, 0
E) -65, -120, 0, -55, -1

In the following code, a multiplication subroutine is written. At the end of the code, the result of 9x3 is stored in register R6.

```
1          push    #9d
2          push    #3d
3          call    #Mul
4          pop     R6
5          (X)
6 fin      jmp     fin
7
8 Mul      push    R15
9          push    R14
10         push    R12
11         mov 8(SP), R15
12         mov #0, R14
13         mov #0, R12
14 add_ag  cmp R12, R15
15         jeq mul_end
16         (Y)
17         inc R12
18         jmp add_ag
19 mul_end mov R14, 10(SP)
20         pop R12
21         pop R14
22         pop R15
23         ret
```

19) Find the correct line for (X).

A) `pop R6`

B) `nop`

C) `add.w #2d, SP`

D) `add.b #2d, SP`

E) `mov 2(SP), R6`

20) Find the correct line for (Y).

A) `add 10(SP), R14`

B) `add 8(SP), R14`

C) `add 8(SP), R10`

D) `add 10(SP), R10`

E) `add 6(SP), R14`