2020

RANK Improvement WORKBOOK

Answer key and Hint of Objective & Conventional *Questions*

Electrical Engineering

Computer Fundamentals

MADE EASY Publications

1 Basic CPU Architecture

LEVEL 1 Objective Questions

1. (d)

2. (b)

3. (d)

LEVEL 2 Objective Questions

4. (c)

5. (c)

6. (d)

LEVEL 3 Conventional Questions

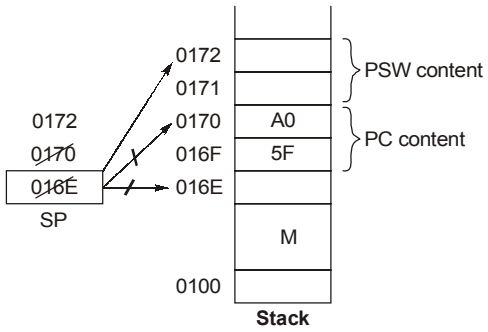
Solution : 1

$$D2 < D < D1$$

Solution : 2

Operation	Instruction size	Required clock cycle
R_0, Memory [5000]	2	2 × 3 + 2 = 8
$R_2 \leftarrow$ Memory [(R_1)]	1	1 × 3 + 2 = 5
$R_2 \leftarrow (R_1 + R_3)$]	1	1 = 1
Memory [6000] $\leftarrow R_2$	2	3 × 2 + 2 = 8
Machine Halt	1	1 + 1 = 2
		Total = 24

Solution : 3



Just before CALL instruction execution, SP contains 016E

While CALL execution:

(*i*) PC contents are pushed i.e., SP incremented by 2 ⇒ SP = 0170

(*ii*) PSW contents are pushed i.e., SP incremented by 2 ⇒ SP = 0172

∴ The value of stack pointer is $(0172)_{16}$.

■■■■

2 Memory Organization

LEVEL 1 Objective Questions

1. (d)
2. (d)
3. (a)
4. (c)
5. (a)

LEVEL 2 Objective Questions

6. (c)
7. (c)
8. (a)

LEVEL 3 Conventional Questions

Solution : 1

For I-cache:

Capacity of tag memory = Tag bits × line size
= 18 bits × 1 K

For D-cache:

Capacity of tag memory = Number of tag bits × Number of set × Number of lines in each set
= 1 K × 19 bits

For L2-cache:

Capacity of tag memory = Number of tag bits × Number of set × Number of lines in each set
= 4 K × 16 bits

Solution : 2

Tag size = No of cache lines*No of bits in tag field for any mapping.

Hence the total size of memory needed at the cache controller to store metadata tags for the write back cache is

= Number of cache lines*(tag bits + valid bit + modified bit)
= 256*(19 + 1 + 1) = 256 * 21
= 5376 bits

Solution : 3

Total time = (3.24 + 1.48) = 4.72 ns

■■■■

3 Input/Output Organization

LEVEL 1 Objective Questions

1. (d)

2. (b)

3. (a)

4. (b)

LEVEL 2 Objective Questions

5. (12.5, 59.60)

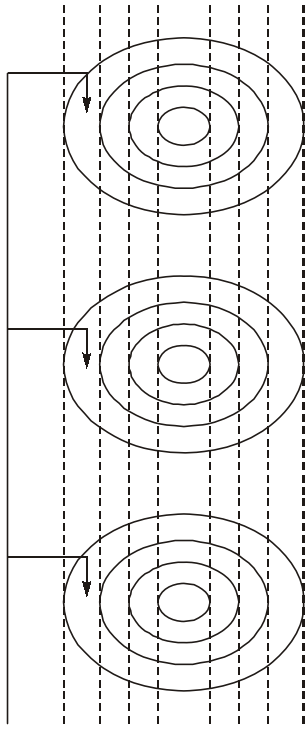
LEVEL 3 Conventional Questions

Solution : 1

Average time to transfer = Average seek time + Average rotational delay + Average data transfer time for transferring 250 bytes

$$= 249.5 + 50 + 0.5 = 300 \text{ ms}$$

Solution : 2



When head is moving then it is reading data from all 16 surface simultaneously.
Currently head is on 9^{th} surface, sector no. 40.
How much data it read from 9^{th} surface = $24 \times 512 = 12288$ bytes.
How much data it read from surface 10^{th} to 15^{th} = $5 \times 512 \times 64 = 163840$ bytes.
How much data it read from cylinder 1201 to 1283

$$82 \times 16 \times 64 \times 512 = 42991616 \text{ bytes}$$

Total data read from 1200^{th} to 1283^{th} cylinder

$$12288 + 163840 + 42991616 = 13167744 \text{ bytes} = 42156 \text{ KB}$$

But we need to real 42797 KB. So we need to go on 1284^{th} cylinder.

Solution : 3

$$\text{Average time} = \text{Average rotational delay} + \text{Average seek time} + \text{Controller's time} + \text{Transfer time} \cong 6.1 \text{ msec}$$

■■■■

4 Data Representation

LEVEL 1 Objective Questions

1. (d)

2. (d)

3. (d)

LEVEL 2 Objective Questions

4. (c)

5. (a)

6. (–64 to +63, 0 to 127)

LEVEL 3 Conventional Questions

Solution : 1

Difference of last two successive numbers = 2^{22}

Solution : 2

1 bit	8 bit	23 bit
0	01111100	1101101000...
S	BE	M

1. Sign = 0 = +ve

2. AE = BE – Bias

BE = 0 1 1 1 1 1 0 0

Bias = 0 1 1 1 1 1 1 1

AE = 1 1 1 1 1 1 0 1

Here sign of AE is negative so take two complement of AE.

i.e., 00000010

1

00000011

⇒ –3

3. Mantissa

∴ Normal Mantissa = 1.M = 1.1101101

Data + 1.1101101×2^{-3} $\{\pm M \times B^{\pm e}\}$

mantissa align to right upto 3 times

↓

+0.0011101101

↓

0.228

■■■■

5 Basic of Operating Systems, Networking and Programming Elements

LEVEL 1 Objective Questions

1. (a)
2. (b)
3. (c)
4. (c)
5. (b)
6. (c)

LEVEL 2 Objective Questions

7. (a)
8. (a)
9. (c)
10. (b)
11. (b)
12. (b)
13. (b)
14. (c)
15. (d)

LEVEL 3 Conventional Questions

Solution : 1

```
#include <stdio.h>
#include <conio.h>
#define ROW 4
#define COL 4
int M[ROW][COL] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16};
main( )
{
        int i, j, t;
        for (i = 0; i < 4; ++i)
        {
                for(j = i; j < 4; ++j)
        {
                t = M[j][i];
                M[i][j] = M[j][i];
                M[j][i] = t;
        }
        }
        for (i = 0; i < 4; ++i)
                for (j = 0; j < 4; ++j)
                   printf (“%d”, M[i][j]);
}
```

Solution : 2

```
#include <stdio.h>
#include <conio.h>
main( )
{
        int  x, y, m, n;
        scanf ("%d %d", &x , &y);
        / * Assume x > 0 and y > 0* /
        m = x; n = y;
        while(m! = n)
        {
              if (m > n)
                 m = m – n;
              else
                   n = n – m;
        }
        printf ("% d", n);
}
```

Solution : 3

```
#include <stdio.h>
#include <conio.h>
void f(int n)
{
        if(n ≤ 1 )
        {
                printf("%d", n);
        }
        else
        {
                f(n/2);
                printf("%d", n%2);
        }
}
```

Solution : 4

At t = 0 only P1 is present so execute it for 1 unit (**remaining CPU time of P1 = 0, now it will perform I/O**).
At t = 1 no process is available for CPU so CPU will remain idle and P1 will perform I/O (**remaining I/O time of P1 = 4**)
At t = 2 P2 arrives and is available for CPU so execute it for 1 unit and simultaneously P1 will do I/O (**remaining CPU time of P2 = 2 & remaining I/O time of P1 = 3**).
At t = 3 P3 also arrives and it is having highest priority among all processes available for CPU so we can execute P3 for its complete CPU burst (i.e 2 unit) as all processes has arrived and simultaneously P1 will do I/O(**remaining CPU time of P3 = 0, now P3 will perform I/O & remaining I/O time of P1 = 1**).
At t = 5 only P2 is available for CPU so it will execute for 1 unit and P1 and P3 will perform I/O(**remaining CPU time of P2 = 1, remaining I/O time of P3 = 2, remaining I/O time of P1 = 0, now P1 will perform CPU**).
At t = 6 P2 and P1 are available for CPU but P1 will be selected because it is having highest priority then P2 so execute P1 for 1 unit(**remaining CPU time of P1 = 2 & remaining I/O time of P3 is 1**)
At t = 7 again P1 will be executed for 1 unit(**remaining CPU time of P1 = 1 & remaining I/O time of P3 = 0, now P3 will perform CPU**).
At t = 8 now P1, P2 & P3 are available for CPU so P3 will be selected based on highest priority and will be executed for 1 unit (**remaining CPU time of P3 = 0, so P3 completed at t = 9**).
At t = 9 P1 will be executed on CPU for 1 unit(**remaining CPU time of P1 = 0, so P1 completed at t = 10**).
At t = 10 only P2 is available for CPU so it will execute its remaining CPU burst (i.e 1 unit) (**remaining CPU time of P2 = 0, now P2 will perform I/O.**)
At t = 11 P2 will perform I/O for 3 units and CPU will remain IDLE.
At t = 14 P2 will perform CPU again and **P2 is completed at t = 15**

P1	IDLE	P2	P3	P2	P1	P1	P3	P1	P2	IDLE	P2
0	1	2	3	5	6	7	8	9	10	11	1415

Solution: 5

A computer network is a group of computer systems and other computing hardware devices that are linked together through communication channels to facilitate communication and resource sharing among a wide range of users.

LAN: Local area network is a computer network which is limited to a small office, single building, multiple buildings inside a campus. LAN is a private network owned and maintained by a single organization.

WAN: A wide area network spans over multiple geographic locations which is composed of multiple LAN's. It is impossible for a small to medium organisation to pull network cables between their two offices in two different countries located thousands of kilometres away. Network service provider provide the connectivity solutions for wide area networks.

LAN:

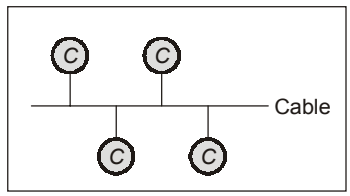
1. It is a private computer network that connects computer in a small physical areas.
2. LAN has higher bandwidth rates. Current LAN's runs on bandwidth of 100 Mbps, 1 Gbps or 10 Gbps.
3. LAN bandwidth rates are almost constant.
4. LAN use ethernet as the LAN standard fast ethernet 100 Mbps or Gigabit ethernet 1/10 Gbps.

WAN

1. WAN is a type of computer network to connect offices which are located in different geographical locations.
2. WAN has lower bandwidth rates compared with LAN. Current WAN runs on bandwidths of 4 Mbps, 8 Mbps, 20 Mbps, 50 Mbps, etc.
3. WAN connectivity solutions are dependent on internet service providers (ISP's)
4. WAN uses technologies like VPN (Vertical Private Network) over internet, frame relay, leased lines as WAN connectivity solutions.

Solution: 6

Computer networks can be broken down historically into topologies, which is a technique of connecting computers.

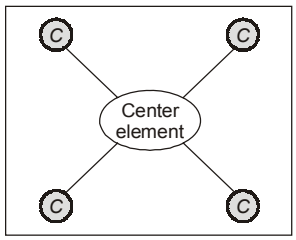


Bus Topology:

- In this design, single cable connects all computers and the information intended for the last node on the network must run through each connected computer.
- If a cable broken, all computers connected down the line cannot reach the network.
- The benefit of a bus topology is a minimal use of cabling.

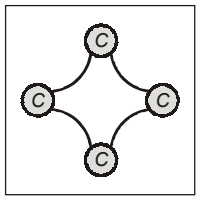
Star Topology:

- In this design of a network, a central node extends a cable to each computer on the network.
- On a star network computers are connected independently to the center of the network.
- If a cable is broken the other computer can operate without problems.
- Star topology requires alot of cabling.



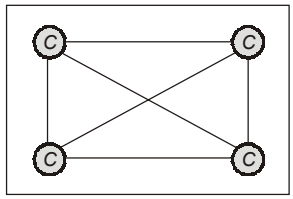
Ring Topology:

- In this design, computers are connected via a single cable, but the end nodes also are connected to each other.
- In this design, the signal circulates through the network until it finds the intended recipient.
- If a network node is not configured properly or it is down temporarily for another reason, the signal will make a number of attempts to find its destination.



Mesh Topology:

- In this design there exists direct and dedicated link between each pair of hosts.
- It is used when high performance and high redundancy is required for a small number of hosts.
- If a network contain '*n*' hosts, each host must have ($n-1$) physical interfaces.



■■■■

