ASSESSMENT I - Model answer

Consider a **Real-Time System** in which there are three tasks. Their period and execution time are as follows:

Processes	Execution time, e	Period, p
P1	35	100
P2	10	50
P3	30	150

The **total utilization of processor** is _____ **%**.

Solution

35/100+10/50+30/150 = 0.35+0.2+ 0.2 = 0.75 = 75%

Assume that there are 4 processes, P1 through P4, and 3 resource types: A, B and C.
At time T0, let consider the following snapshot of the system:

Process	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P1	0	1	0	7	5	5	2	3	0
P2	3	0	2	3	2	2			
P3	3	0	2	9	0	2			
P4	2	1	1	2	2	2			

The system is currently in a safe state.
What is the execution order of the processes so that the system remains in a safe state?

a. P2 P4 P3 P1
b. P2 P1 P3 P4

c. P1 P2 P3 P4
d. P1 P3 P2 P4
e. P3 P1 P4 P2
f. P4 P1 P2 P3
g. P3 P1 P2 P4
h. P4 P3 P1 P2

Solution

Total resource (A B C) = (10 5 5)

Process	Need		
	A	B	C
P1	7	4	5
P2	0	2	0
P3	6	0	0
P4	0	1	1

P2: (2 3 0) + (3 0 2) = (5 3 2)
P4: (2 1 1) + (5 3 2) = (7 4 3)
P3: (3 0 2) + (7 4 3) = (10 4 5)
P1: (0 1 0) + (10 4 5) = (10 5 5)

Calculate the predicted burst time using exponential averaging for the **fifth** process if the predicted burst time for the first process is **10** units and actual burst time of the first four processes is **2, 4, 6** and **8** units respectively, given **α = 0.5**.
The scheduling algorithm is the Shortest Job First (SJF).

Solution

Predicted burst time for **1st process** = **10 units**
Actual burst time of the first four processes = 2, 4, 6, 8 α = 0.5

Predicted burst time for **2nd process** = α x Actual burst time of 1st process + (1-α) x Predicted burst time for 1st process

= 0.5 x 2 + 0.5 x 10 = 1 + 5 = 6 units

Predicted burst time for **3rd process** = α x Actual burst time of 2nd process + (1-α) x Predicted burst time for 2nd process

= 0.5 x 4 + 0.5 x 6 = 2 + 3 = 5 units

Predicted burst time for **4th process** = α x Actual burst time of 3rd process + (1-α) x Predicted burst time for 3rd process

= 0.5 x 6 + 0.5 x 5 = 3 + 2.5 = 5.5 units

Predicted burst time for **5th process** = α x Actual burst time of 4th process + (1-α) x Predicted burst time for 4th process

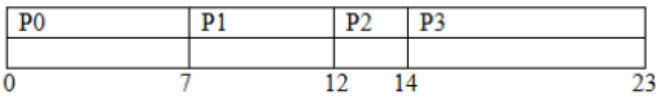
= 0.5 x 8 + 0.5 x 5.5 = 4 + 2.75 = **6.75 units**

Consider the following scenario of processes and the **FCFS (first-come first-served)** scheduling algorithm.

Process	Burst time
P0	7
P1	5
P2	2
P3	9

Calculate **the average waiting time** of the system.

Solution



The waiting time of P0 = 0
The waiting time of P1 = 7
The waiting time of P2 = 12
The waiting time of P3 = 14
The average waiting time = (0 + 7 + 12 + 14) /4 = 33/4 = 8.25

You are interested in researching how fast a computer virus spreads in a network and want to write a program to simulate it.
Your program should first read an integer representing the ***total computers*** of the network, followed by an integer representing the number of computers each ***newly infected computer*** contaminates.
Assume that **one** computer was infected on **day 1**.
You must calculate and display the day on which the ***entire*** computers of the network will be infected.

Test case 1:
Input:
5 2
Output:
3

Test case 2:
Input:
10 2
Output:
4

Test case 3:
Input:
50 3
Output:
5

Solution

```
#include <stdio.h>
int main() {
    int calc, infect;
    scanf("%d %d", &calc, &infect);
    int day = 1;
    int newInfected = 1;
    int totalInfected = 1;
    while (totalInfected < calc) {
        day++;
        newInfected = infect * newInfected;
        totalInfected = totalInfected + newInfected;
    }
    printf("%d\n", day);
    return 0;
}
```

	Input	Expected	Got	
✔	5 2	3	3	✔
✔	10 2	4	4	✔
✔	50 3	5	5	✔

Passed all tests! ✔

Question author's solution (C):

```c
#include <stdio.h>
int main() {
int calc, infect;
scanf("%d %d", &calc, &infect);
int day = 1;
int newInfected = 1;
int totalInfected = 1;
while (totalInfected < calc) {
day++;
newInfected = infect * newInfected;
totalInfected = totalInfected + newInfected;
}
printf("%d\n", day);
return 0;
}
```