

OS Lab Assignment :

Name : Asad Yousuf

Roll No : BTECH10072/20

Section : IT

Question 1 : FCFS Algorithm

```
#include <stdio.h>
int main(){
    printf("Process    Burst Time ");
    printf("\n");
    int p[]={1,2,3};
    int size=sizeof(p) / sizeof p[0];
    int burst[]={24,3,3};
    int wait[size];
    int avg[size];
    int tot_wt = 0,tot_tat = 0;

    wait[0] = 0;
    for (int i=1; i<size; i++){
        wait[i] = wait[i-1]+burst[i-1] ;
    }
    for (int i = 0; i < size ; i++){
        avg[i] = burst[i] + wait[i];
    }
    for (int i=0; i<size; i++)
    {
        tot_wt = tot_wt + wait[i];
        tot_tat = tot_tat + avg[i];
    }
    int s=(float)tot_wt / (float)size;
    int t=(float)tot_tat / (float)size;
```

```

printf("Average waiting time = %d",s);
printf("\n");
printf("Average waiting time for P1= %d ",t);
}

```

The screenshot shows a C++ IDE with a file named `main.c`. The code implements a program to calculate the average waiting time for processes. The console output shows the results of the program execution.

```

1 #include <stdio.h>
2 int main(){
3     printf("Process    Burst Time ");
4     printf("\n");
5     int p[]={1,2,3};
6     int size=sizeof(p) / sizeof p[0];
7     int burst[]={24,3,3};
8     int wait[size];
9     int avg[size];
10    int tot_wt = 0,tot_tat = 0;
11
12    wait[0] = 0;
13    for (int i=1; i<size; i++){
14        wait[i] = wait[i-1]+burst[i-1] ;
15    }
16    for (int i = 0; i < size ; i++){
17        avg[i] = burst[i] + wait[i];
18    }
19    for (int i=0; i<size; i++)
20    {
21        tot_wt = tot_wt + wait[i];
22        tot_tat = tot_tat + avg[i];
23    }
24    int s=(float)tot_wt / (float)size;
25    int t=(float)tot_tat / (float)size;
26    printf("Average waiting time = %d",s);
27    printf("\n");
28    printf("Average waiting time for P1= %d ",t);
29 }
30
31
32

```

```

> make -s
> ./main
Process    Burst Time
Average waiting time = 17
Average waiting time for P1= 27 >

```

Question 2 : SJF Algorithm

```

#include<stdio.h>
void waitingTime(int n,int burst_time[],int wt[]){
    wt[0]=0;

    for(int i=1;i<n;i++){
        wt[i] = burst_time[i-1] + wt[i-1];
    }
}

void findAvgTime(int n,int process[],int burst_time[]){
    int wt[n];
    waitingTime(n,burst_time, wt);
    int totalWaitingTime = 0;
    for(int i=0;i<n;i++){
        totalWaitingTime+=wt[i];
    }
    float avg = totalWaitingTime/n;
    printf("Average waiting time = %f",avg);
}

int main(){
    int process[] = {1,2,3};
    int n = sizeof process/ sizeof process[0];
    int burst_time[] = {24,3,3};
    findAvgTime(n,process,burst_time);
}

```

```
    return 0;  
}
```

The screenshot shows a C code editor with a dark theme. The left sidebar displays the 'Files' panel with a project named 'main.c'. The main editor area shows the source code for 'main.c', which includes a function to calculate waiting times and a main function that uses it. The right sidebar shows the 'Console' panel with the output of the program.

```
1 #include<stdio.h>  
2 void waitingTime(int n,int burst_time[],int wt[]){  
3     wt[0]=0;  
4  
5     for(int i=1;i<n;i++){  
6         wt[i] = burst_time[i-1] + wt[i-1];  
7     }  
8 }  
9 void findAvgTime(int n,int process[],int burst_time[]){  
10     int wt[n];  
11     waitingTime(n,burst_time, wt);  
12     int totalWaitingTime = 0;  
13     for(int i=0;i<n;i++){  
14         totalWaitingTime+=wt[i];  
15     }  
16     float avg = totalWaitingTime/n;  
17     printf("Average waiting time = %f",avg);  
18 }  
19  
20 int main(){  
21     int process[] = {1,2,3};  
22     int n = sizeof process/ sizeof process[0];  
23     int burst_time[] = {24,3,3};  
24     findAvgTime(n,process,burst_time);  
25     return 0;  
26 }
```

The console output shows the following commands and results:

```
> make -s  
> ./main  
Average waiting time = 17.000000
```