Ex. No.: 6a) 22/2/25
Date:

FIRST COME FIRST SERVE

Aim:

To implement First-come First- serve (FCFS) scheduling technique

Algorithm:

1. Get the number of processes from the user.
2. Read the process name and burst time.
3. Calculate the total process time.
4. Calculate the total waiting time and total turnaround time for each process 5. Display the process name & burst time for each process. 6. Display the total waiting time, average waiting time, turnaround time

Program Code:

```
n=int(input("Enter the no. of process "))
l=[]
s=0
s1=0
for i in range(n):
    print("Enter the brust time of process", i,":")
    n1=int(input())
    l.append(n1)
    s+=n1
    print("Turn around time", i,":", s)
    print("Waiting time of process", i,":", s-n1)
    print()
print("Average Turn around Time ", s/n)
print("Average Waiting Time ", (s-n1)/n)
```


PDF Scanner ACE Scanner

Output

Enter no. of process: 3

Enter brust time of Process 0:

5

Turn around Time 0: 5

Waiting time of process 0: 0

Enter the brust time of Process 1:

4

Turn around time 1: 9

Waiting time of process 1: 5

Enter the brust time of Process 2:

3

Turn around time of Process 2: 12

Waiting time of Process 2 : 9

Average turn around time : 4

Average waiting time : 3

PDF Scanner ACE Scanner

Sample Output:
Enter the number of process:
3
Enter the burst time of the processes:
24 3 3

Process	Burst Time	Waiting Time	Turn Around Time
0	24	0	24
1	3	24	27
2	3	27	30

Average waiting time is: 17.0
Average Turn around Time is: 19.0

Result:

Thus the python program for FCFS Scheduling technique is successfully executed

PDF Scanner ACE Scanner

Ex. No.: 6b) 28/2/25
Date:

SHORTEST JOB FIRST

Aim:

To implement the Shortest Job First (SJF) scheduling technique

Algorithm:

1. Declare the structure and its elements.
2. Get number of processes as input from the user.
3. Read the process name, arrival time and burst time
4. Initialize waiting time, turnaround time & flag of read processes to zero. 5. Sort based on burst time of all processes in ascending order 6. Calculate the waiting time and turnaround time for each process. 7. Calculate the average waiting time and average turnaround time. 8. Display the results.

Program Code:

```
n=int(input("Enter no. of Process"))
l=[]
for i in range(n):
    print("Enter the Brust time of the Process",i)
    n1=int(input())
    l.append(n1)

l1=l.copy()
l1.sort()
s=0
d={}
print()
for i in range(n):
    d[l.index(l1[i])]=[s,s+l1[i]]
    s+=l1[i]
```

Output

Enter no. of process 5
Enter the Brust time of the Process 0
4
Enter the Brust time of the Process 1
6
Enter the Brust time of the Process 2
2
Enter the Brust time of the Process 3
1
Enter the Brust time of the process 4
9

PROCESS 0

The waiting time of process 0 : 3
The Turn around Time of process 0 : 7

PROCESS 1

The waiting time of process 1 : 7
The Turn around Time of process 1 : 13

PROCESS 2

The waiting time of process 2 : 1
The Turn around time of process 2 : 3

Process 3

The waiting time of process 3 : 0
The Turn around time of process 3 : 1

PROCESS 4

The waiting time of process 4 : 13
The turn around Time of process 4 : 22

PDF Scanner ACE Scanner

```
s=0
s1=0
for i in range(n):
        print("\t\t\t\t PROCESS", i)
        print("The Waiting Time Of Process",i," :",d[i][0])
        st=d[i][0]
        s1+=d[i][1]
        print("The Turn around Time of Process",i," :",d[i][1])
        print()
print("Average waiting Time",s/n)
print("Average Turn around Time",s1/n)
```


PDF Scanner ACE Scanner

Average waiting time 4.8

Average Turn around Time 9.2

PDF Scanner ACE Scanner

Sample Output:
Enter the number of process:
4
Enter the burst time of the processes:
8 4 9 5

Process	Burst Time	Waiting Time	Turn Around Time
2	4	0	4
4	5	4	9
1	8	9	17
3	9	17	26

Average waiting time is: 7.5
Average Turn Around Time is: 13.0

Result:

Thus the python program SJF scheduling technique is successfully executed

PDF Scanner ACE Scanner

Ex. No.: 6c)
Date: 21/3/25

PRIORITY SCHEDULING

Aim:

To implement priority scheduling technique

Algorithm:

1. Get the number of processes from the user.
2. Read the process name, burst time and priority of process.
3. Sort based on burst time of all processes in ascending order based priority 4. Calculate the total waiting time and total turnaround time for each process 5. Display the process name & burst time for each process.
6. Display the total waiting time, average waiting time, turnaround time

Program Code:

```
n=int(input("Enter the no. of process"))
l=[]
for i in range(n):
    print("Enter brust time of the Process",i)
    n1=int(input())
    print("Enter the priority of the Process",i)
    n2=int(input())
    l.append([n1,n2])
l1=l.copy()
for i in range(n-1):
    for j in range(n-1-i):
        if l1[j][1]>l1[j+1][1]:
            temp=l1[j]
            l1[j]=l1[j+1]
            l1[j+1]=temp
```

Enter the no. of process 3
Enter brust time of the Process 0
5
Enter ~~bre~~ the priority of the Process 0
3
Enter brust time of the Process 1
4
Enter the priority of the Process 1
2
Enter brust time of the Process 2
3
Enter the Priority of the Process 2
1

PROCESS 0

The waiting time of the Process 0: 7
The Turn around Time of the Process 0: 12

PROCESS 1

The waiting time of the Process 1: 3
The Turn around time of the Process 1: 7

PROCESS 2

The waiting time of the Process 2: 0
The Turn around Time of Process 2: 3

The Average Waiting Time: 3.333
The Average Turn around Time: 7.333

PDF Scanner ACE Scanner

```
s=0
d={}
for i in range(n):
        d[l.index(l1[i])] = [s, s+l1[i][0]]
        s+=l1[i][0]
s=0
s1=0
for i in range(n):
        print("\t\t\t PROCESS",i)
        print("\t The waiting Time of Process",i,":",d[i][0])
        s+=d[i][0]
        print("\t The Turn around time of Process",i,":",
                                                d[i][1])
        s1+=d[i][1]

print("The Average Waiting time:",s/n)
print("The Average Turn around time:",s1/n)
```


PDF Scanner ACE Scanner

Sample Output:

```
C:\Users\admin\Desktop\Untitled1.exe
Enter Total Number of Process:4

Enter Burst Time and Priority

P[1]
Burst Time:6
Priority:3

P[2]
Burst Time:2
Priority:2

P[3]
Burst Time:14
Priority:1

P[4]
Burst Time:6
Priority:4

Process      Burst Time          Waiting Time     Turnaround Time
P[3]              14                  0                14
P[2]              2                   14               16
P[1]              6                   16               22
P[4]              6                   22               28

Average Waiting Time=13
Average Turnaround Time=20
```

Result:

Thus the python program for priority scheduling technique is successfully executed

PDF Scanner ACE Scanner

Ex. No.: 6d)
Date 28/3/25

ROUND ROBIN SCHEDULING

Aim:

To implement the Round Robin (RR) scheduling technique

Algorithm:

1. Declare the structure and its elements.
2. Get number of processes and Time quantum as input from the user.
3. Read the process name, arrival time and burst time
4. Create an array **rem_bt[]** to keep track of remaining burst time of processes which is initially copy of bt[] (burst times array)
5. Create another array **wt[]** to store waiting times of processes. Initialize this array as 0. 6. Initialize time : t = 0
7. Keep traversing the all processes while all processes are not done. Do following for i'th process if it is not done yet.
   a- If rem_bt[i] > quantum
   (i) t = t + quantum
   (ii) bt_rem[i] -= quantum;
   b- Else // Last cycle for this process
   (i) t = t + bt_rem[i];
   (ii) wt[i] = t - bt[i]
   (iii) bt_rem[i] = 0; // This process is over
8. Calculate the waiting time and turnaround time for each process.
9. Calculate the average waiting time and average turnaround time.
10. Display the results.

Program Code:

```
n=int(input("Enter the no. of process"))
l=[]
t=int(input("Enter the Quantum "))
c1=0
for i in range(n):
    print("Enter the process", i, "Brust time:")
    n1=int(input())
    l.append(n1)
    c1+=n1
l1=l.copy()
l2=[]
l3=[]
```


PDF Scanner ACE Scanner

Enter the no. of process 4

Enter the Quantum 2

Enter the process 0 Brust time

6

Enter the process 1 Brust time

5

Enter the process 2 Brust time

4

Enter the process 3 Brust time

3

PROCESS 0

Turn around Time of Process 0 : 14

Waiting time of Process 0 : 11

PROCESS 1

Turn around time of Process 1 : 18

Waiting Time of Process 1 : 13

PROCESS 2

Turn around Time of Process 2 : 14

Waiting Time of Process 2 : 10

PROCESS 3

Turn around Time of Process 3 : 15

Waiting Time of Process 3 : 12

Average Turn around time : 16.00

Average waiting time : 11.5

PDF Scanner ACE Scanner

```
i=0
c=0
while (c!=c1):
        if(l1[i] >=t):
            l3.append(i)
            l3.append(c+t)
            c+=t
            l1[i]-=t
        elif (l[i] < t):
                l2.append(i)
                l3.append(c+l1[i])
                c+=l1[i]
                l1[i]-=1
        else:
            continue
        i=(i+1)%n
s=0
s1=0
for i in range(n):
    print("\t\t PROCESS ",i)
    print("Turn around time of Process",i,l3[-1-l2[::-1].index(i)])
    s+=l3[-1-l2[::-1].index(i)])
    print("Waiting time of process ",i,l3[-1-l2[::-1].index(i)]-l[i])
    s1+=(l3[-1-l2[::-1].index(i)]-l[i])
    print()

print("Average Turn around time : s/n)
print("Average waiting time : s1/n)
```

Result:

Thus the python program for round robin scheduling technique is successfully executed.

PDF Scanner ACE Scanner