Operating System Assignment On Shortest Job First(SJF)

Student Name:Harish CH

Student ID: 11703388

Roll no:59

Email Address: chappidi100@gmail.com

GitHub Link: https://github.com/HARISHCH684/harish

Code: 17

Submitted To: Ms.shaina Gupta

Description:

This project is based on designing a scheduler following non preemptive scheduling approach to schedule the processes that arrives at different units and having burst time double the arrival time. We are going to find the average waiting time if Scheduler selects the process with largest burst time from the queue for execution when process is not being preempted until it finishes its service time. We are also going to find the average time when the processes are executed accordiing to SJF approach with the same attribute values.

Algorithm:

```
#include<stdio.h>
#include<conio.h>
int main()
{
	int bt[10],p[10],n,temp,i,j,wt[10],sum=0;
	float avg;
	printf("Enter total no of process:");
	scanf("%d",&n);
	printf("\n Enter burst time for each process:-");
	for(i=0;i<n;i++)
	{
		printf("\n Burst time of process P%d:",i);
		scanf("%d",&bt[i]);
		p[i]=i;

	}
	printf("\n\n scheduler will execute the process which has the highest burst time\n\n");
	for(i=0;i<n-1;i++)
	{
		for(j=i+1;j<n;j++)
		{
			if(bt[i]<bt[j])
```

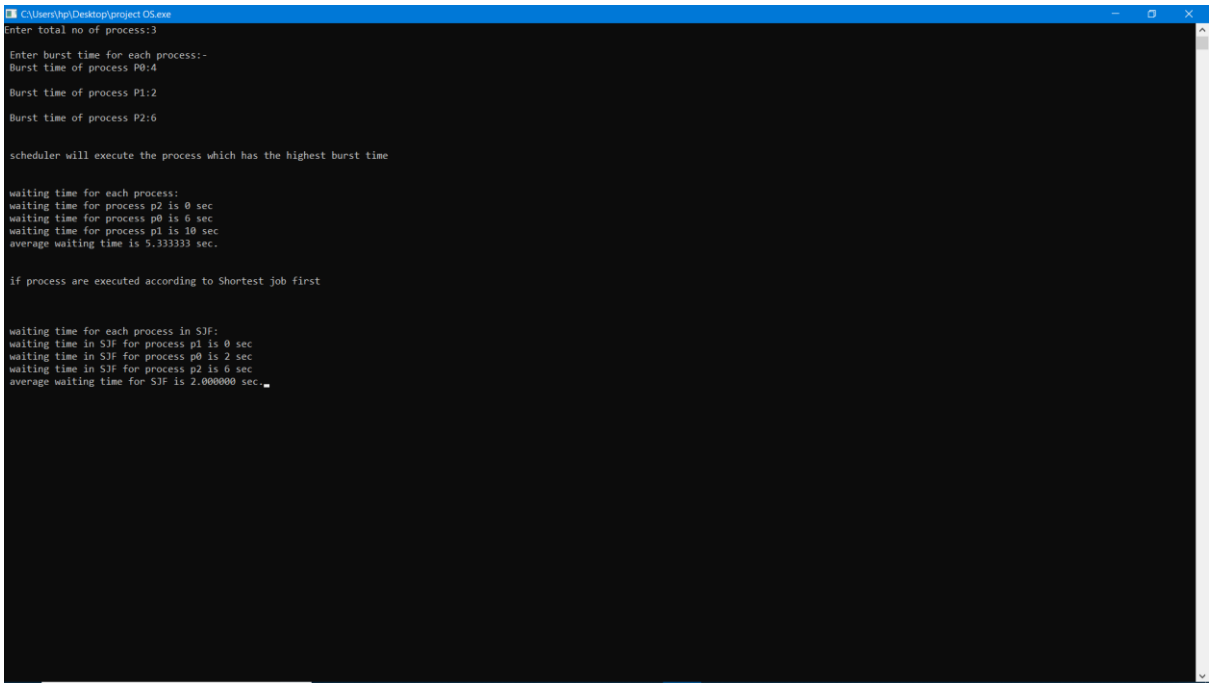
```
                {
                        temp=bt[i];
                        bt[i]=bt[j];
                        bt[j]=temp;
                        temp=p[i];
                        p[i]=p[j];
                        p[j]=temp;
                }
        }
}
wt[0]=0;
for(i=1;i<n;i++)
{
        wt[i]=wt[i-1]+bt[i-1];
}
for(i=0;i<n;i++)
{
        sum+=wt[i];

}
avg=(float)sum/n;
printf("\n waiting time for each process:");
for(i=0;i<n;i++)
{
        printf("\n waiting time for process p%d is %d sec",p[i],wt[i]);
}
printf("\n average waiting time is %f sec.",avg);
printf("\n\n\n if process are executed according to Shortest job first\n\n\n");
for(i=0;i<n-1;i++)
{
```

```
        for(j=i+1;j<n;j++)
        {
                if(bt[i]>bt[j])
                {
                        temp=bt[i];
                        bt[i]=bt[j];
                        bt[j]=temp;
                        temp=p[i];
                        p[i]=p[j];
                        p[j]=temp;
                }
        }
}
wt[0]=0;
for(i=1;i<n;i++)
{
        wt[i]=wt[i-1]+bt[i-1];
}
for(i=0;i<n;i++)
{
        sum=0;
        sum+=wt[i];

}
avg=(float)sum/n;
printf("\n waiting time for each process in SJF:");
for(i=0;i<n;i++)
{
        printf("\n waiting time in SJF for process p%d is %d sec",p[i],wt[i]);
}
```

```
    printf("\n average waiting time for SJF is %f sec.",avg);
    getch();
    return 0;
}
```



input:

Total no of processes=3

Burst time for process p0=4

Burst time for process p1=2

Burst time for process p2=6

Output:

Scheduler will execute the process which has the highest burst time:

Waiting time for each process:

Waiting time for process p2=0 sec

Waiting time for process p0=6 sec

Waiting time for process p1=10 sec

Average waiting time is 5.333333 sec

If the process are executed according to Shortest job first:

Waiting time for each process in SJF:

Waiting time in SJF for process p1=0 sec

Waiting time in SJF for process p0=2 sec

Waiting time in SJF for process p2=6 sec

Average waiting time for SJF is 2.000000 sec.

GIT HUB LINK: https://github.com/HARISHCH684/harish