Experiment No # 07
Experiment Name # Shortest Job First (SJF) Algorithm.

SJF:

1. Best approach to minimize waiting time.

2. Easy to implement in Batch systems where required CPU time is known in advance.

3. Impossible to implement in interactive systems where required CPU time is not known.

4. The processer should know in advance how much time process will take.

Code:

```c
#include<stdio.h>
void main()
{
        int bt[20],p[20],wt[20],tat[20],i,j,n,total=0,pos,temp;
        float avg_wt,avg_tat;
        printf("Enter number of process: ");
        scanf("%d",&n);
        printf("\nEnter Burst Time: \n");
        for(i=0;i<n;i++){
            printf("p%d: ",i+1);
            scanf("%d",&bt[i]);
            p[i]=i+1;
        }
        for(i=0;i<n;i++){
            pos=i;
            for(j=i+1;j<n;j++){
                if(bt[j]<bt[pos])
                pos=j;
            }
            temp=bt[i];
            bt[i]=bt[pos];
            bt[pos]=temp;

            temp=p[i];
            p[i]=p[pos];
            p[pos]=temp;
        }
      wt[0]=0;
      for(i=1;i<n;i++){
            wt[i]=0;
            for(j=0;j<i;j++)
                wt[i]+=bt[j];
                total+=wt[i];
      }
      avg_wt=(float)total/n;
```

```
    total=0;
    printf("\nProcess\t   Burst Time    \tWaiting Time\tTurnaround Time");
    for(i=0;i<n;i++){
            tat[i]=bt[i]+wt[i];
            total+=tat[i];
        printf("\np%d\t\t  %d\t\t   %d\t\t\t%d",p[i],bt[i],wt[i],tat[i]);
    }
    avg_tat=(float)total/n;
    printf("\nAverage Waiting Time=%f",avg_wt);
    printf("\nAverage Turnaround Time=%f\n",avg_tat);
}
```

Output: