OS Lab

Aadhitya Swarnesh I

6, Mar 2020

```
// implementation of FCFS
// scheduling
#include<stdio.h>
#include<stdlib.h>
#include<stdbool.h>
//---------------------------------
int turnarroundtime(int processes[], int n,
int bt[], int wt[], int tat[]) {

   for (int i = 0; i < n ; i++)
   tat[i] = bt[i] + wt[i];
   return 1;
}
int waitingtime(int processes[], int n,
int bt[], int wt[], int quantum) {

   int rem_bt[n];
   for (int i = 0 ; i < n ; i++)
   rem_bt[i] = bt[i];
   int t = 0;
   while (1) {
      _Bool done = true;
      for (int i = 0 ; i < n; i++) {
         if (rem_bt[i] > 0) {
            done = false;
            if (rem_bt[i] > quantum) {
               t += quantum;
               rem_bt[i] -= quantum;
            }
            else {
               t = t + rem_bt[i];
               wt[i] = t - bt[i];
               rem_bt[i] = 0;
            }
         }
      }
      if (done == true)
         break;
   }
   return 1;
}
int findavg(int processes[], int n, int bt[],
int quantum) {
   int wt[n], tat[n], total_wt = 0, total_tat = 0;
   waitingtime(processes, n, bt, wt, quantum);
   turnarroundtime(processes, n, bt, wt, tat);
   printf("Processes \t\t Burst time \t\t Waiting time \t\t Turn around time\n");
   for (int i=0; i<n; i++) {
      total_wt = total_wt + wt[i];
```

```
      total_tat = total_tat + tat[i];
      printf("\t%d\t\t\t%d\t\t\t%d\t\t\t%d\n",i+1, bt[i], wt[i], tat[i]);
   }
   printf("Average waiting time = %f", (float)total_wt / (float)n);
   printf("\nAverage turnaround time = %f\n", (float)total_tat / (float)n);
   return 1;
}
//--------------------
void findWaitingTime(int processes[], int n,int bt[], int wt[])
{
    wt[0] = 0;

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

void findTurnAroundTime( int processes[], int n,
             int bt[], int wt[], int tat[])
{
    for (int i = 0; i < n ; i++)
        tat[i] = bt[i] + wt[i];
}

void findavgTime( int processes[], int n, int bt[])
{
    int wt[n], tat[n], total_wt = 0, total_tat = 0;


    findWaitingTime(processes, n, bt, wt);
    findTurnAroundTime(processes, n, bt, wt, tat);
    printf("Processes \t\t Burst time \t\t Waiting time \t\t Turn around time\n");

    for (int i=0; i<n; i++)
    {
        total_wt = total_wt + wt[i];
        total_tat = total_tat + tat[i];
        printf(" %d ",(i+1));
        printf("      %d ", bt[i] );
        printf("      %d",wt[i] );
        printf("      %d\n",tat[i] );
    }
    int s=(float)total_wt / (float)n;
    int t=(float)total_tat / (float)n;
    printf("Average waiting time = %d",s);
    printf("\n");
    printf("Average turn around time = %d \n",t);
}


int main()
{

        printf(" 1) FCFS \n 2) SJF \n 3) Priority Scheduling \n 4) RR algorithm \n");
        printf("Enter your choice ");
        int ch; scanf("%d",&ch);


        if(ch==1)
        {
        int processes[] = { 1, 2, 3};
```

```
    int n = sizeof processes / sizeof processes[0];


    int burst_time[] = {10, 5, 8};

    findavgTime(processes, n, burst_time);

     }
  else if(ch==2)
{

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("\nProcesses \t\t Burst time \t\t Waiting time \t\t Turn around time\n");
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    %d",p[i],bt[i],wt[i],tat[i]);
    }

    avg_tat=(float)total/n;
    printf("\n\nAverage Waiting Time=%f",avg_wt);
    printf("\nAverage Turnaround Time=%f\n",avg_tat);
  }

else if(ch==3)
{
    int bt[20],p[20],wt[20],tat[20],pr[20],i,j,n,total=0,pos,temp,avg_wt,avg_tat;
    printf("Enter Total Number of Process:");
    scanf("%d",&n);

    printf("\nEnter Burst Time and Priority\n");
    for(i=0;i<n;i++)
    {
        printf("\nP[%d]\n",i+1);
        printf("Burst Time:");
        scanf("%d",&bt[i]);
        printf("Priority:");
        scanf("%d",&pr[i]);
        p[i]=i+1;
    }

    for(i=0;i<n;i++)
    {
        pos=i;
        for(j=i+1;j<n;j++)
        {
            if(pr[j]<pr[pos])
                pos=j;
        }

        temp=pr[i];
        pr[i]=pr[pos];
        pr[pos]=temp;

        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=total/n;
```

```
    total=0;

    printf("\nProcesses \t\t Burst time \t\t Waiting time \t\t Turn around time\n");
    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=total/n;
    printf("\n\nAverage Waiting Time=%d",avg_wt);
    printf("\nAverage Turnaround Time=%d\n",avg_tat);

}
else if(ch==4) // RR
{
    int processes[] = { 1, 2, 3};
   int n = sizeof processes / sizeof processes[0];
   int burst_time[] = {8, 6, 12};
   int quantum = 2;
   findavg(processes, n, burst_time, quantum);
   return 0;
}

}
```

This code has been placed in the file called as “all_programs.c”

```
(base) Aadhityas-MacBook-Air:Desktop aadhitya$ gcc all_programs.c
(base) Aadhityas-MacBook-Air:Desktop aadhitya$ ./a.out
 1) FCFS
 2) SJF
 3) Priority Scheduling
 4) RR algorithm
Enter your choice 4
Processes               Burst time              Waiting time            Turn around time
        1                       8                       12                      20
        2                       6                       10                      16
        3                       12                      14                      26
Average waiting time = 12.000000
Average turnaround time = 20.666666
(base) Aadhityas-MacBook-Air:Desktop aadhitya$
```