OPERATING SYSTEM PROJECT

Name: DEEPU KUMAR.

Reg. No. : 11616204

Section : K1653

Email Id : erdeepukumar@gmail.com

GitHub Link: https://github.com/Deepudeepak76/K1653_B41_DEEPUDEEPAK

Question No. : 9

Problem:

Design a scheduler that uses a preemptive priority scheduling algorithm based on dynamically changing priority. Larger number for priority indicates higher priority. When the process starts execution (i.e. CPU assigned), priority for that process changes at the rate of m=1.When the process waits for CPU in the ready queue (but not yet started execution), its priority changes at a rate n=2. All the processes are initially assigned priority value of 0 when they enter ready queue for the first time. The time slice for each process is q = 1. When two processes want to join ready queue simultaneously, the process which has not executed recently is given priority. I have also calculated the average waiting time for each process and my program is generic i.e. number of processes, their burst time and arrival time will be entered by user.

ALGORITHM:

1. Declare a structure process (It is collection of all the necessary attribute a process contain like service time, Arrival time, Priority, Process ID, Processed flag etc.)

2. A function processCreation() (It asked user about the number of process and enter details about it like Service time, arrival time, Process ID, and also assign initial priority to 0)

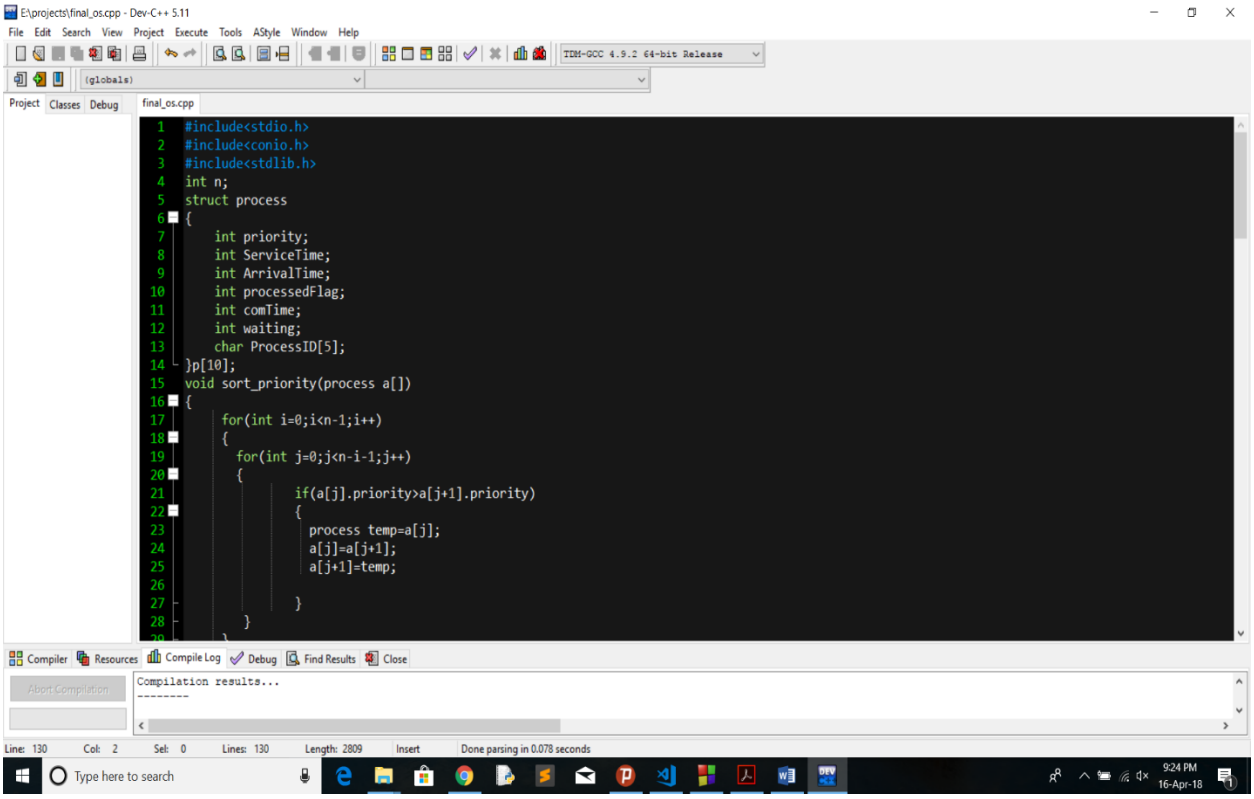
3. A function sortProcess() (It sort all the process according to their arrival time)

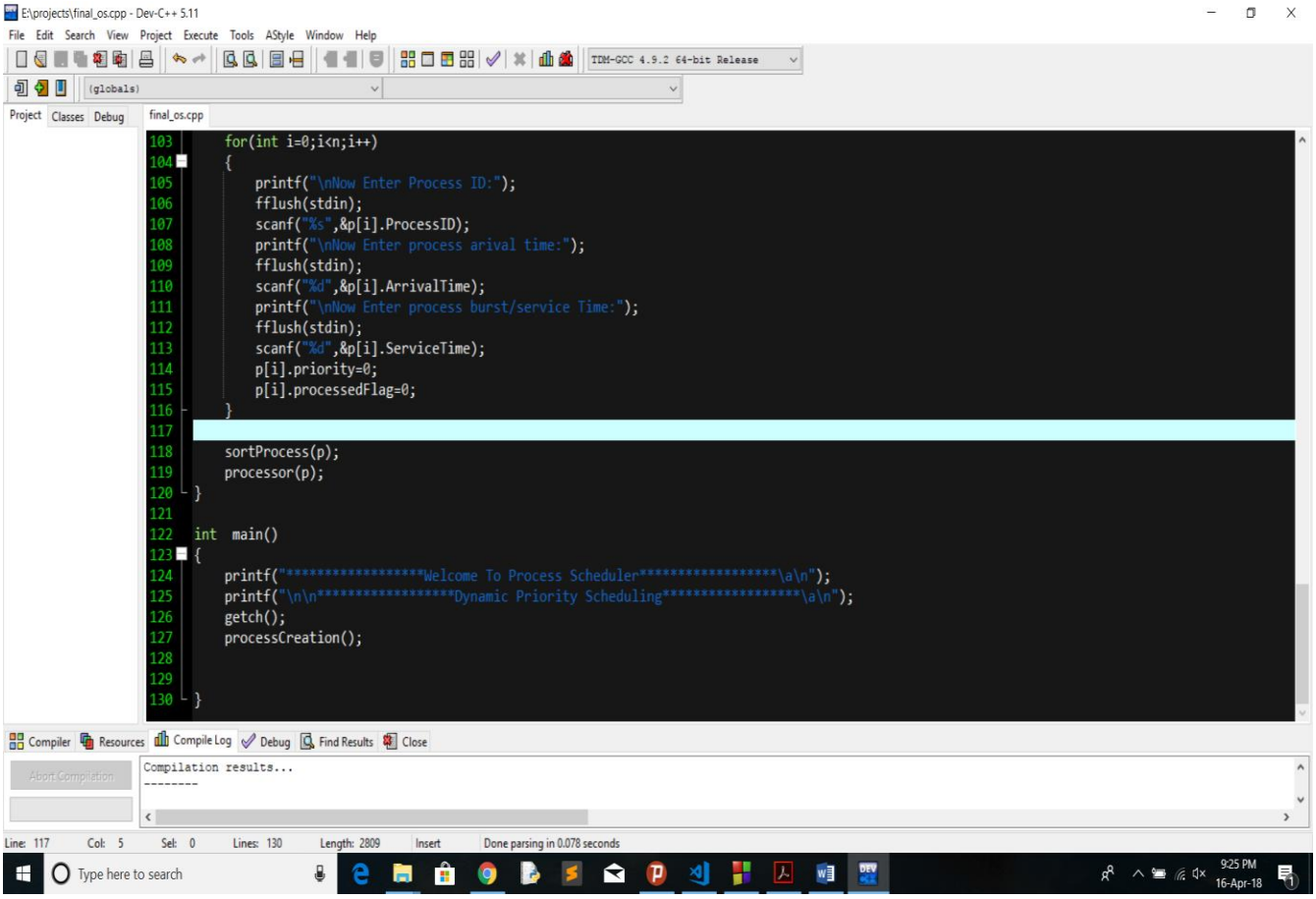
4. A function processor() (It does all the process execution, it responsible for processing or jobs displaying all the jobs and waitng time.

Time Complexity:

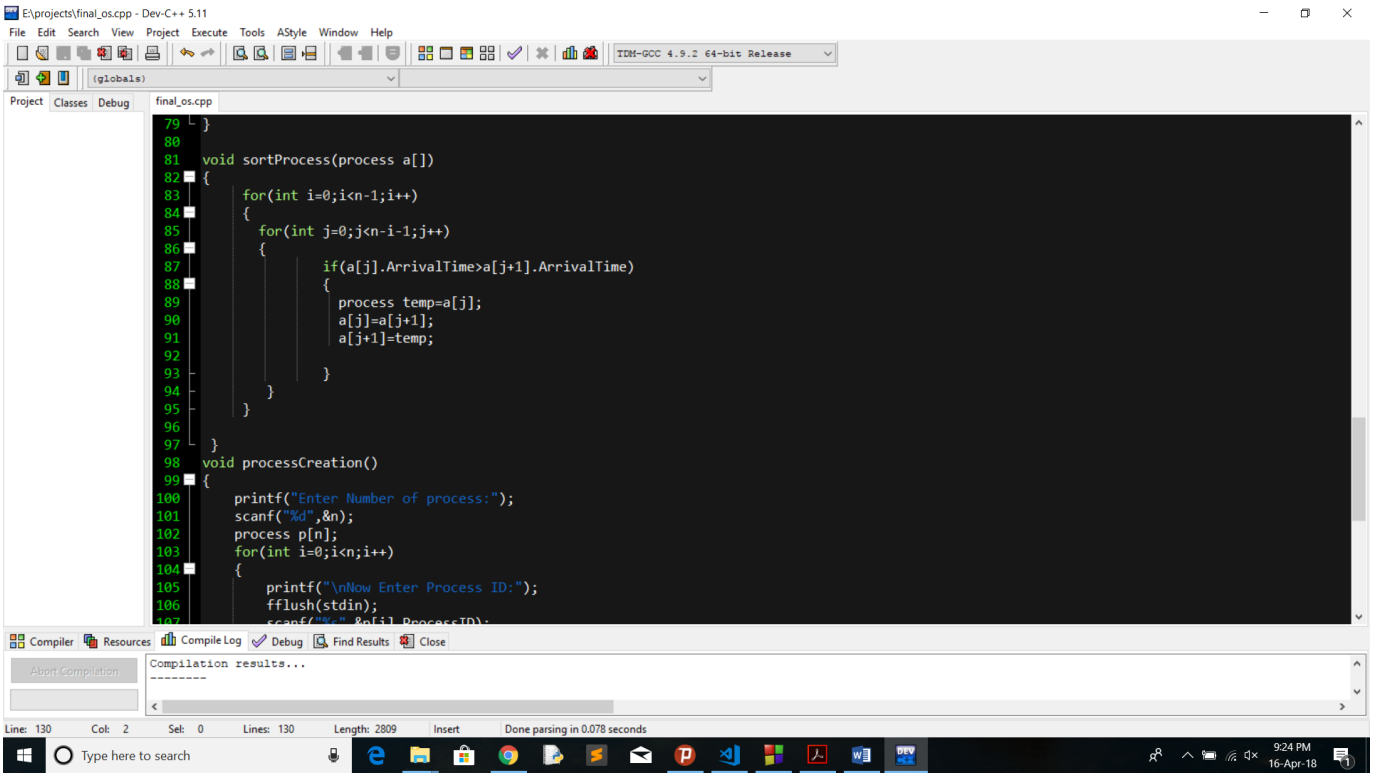
O(nlogn)

CODE SNIPPET:

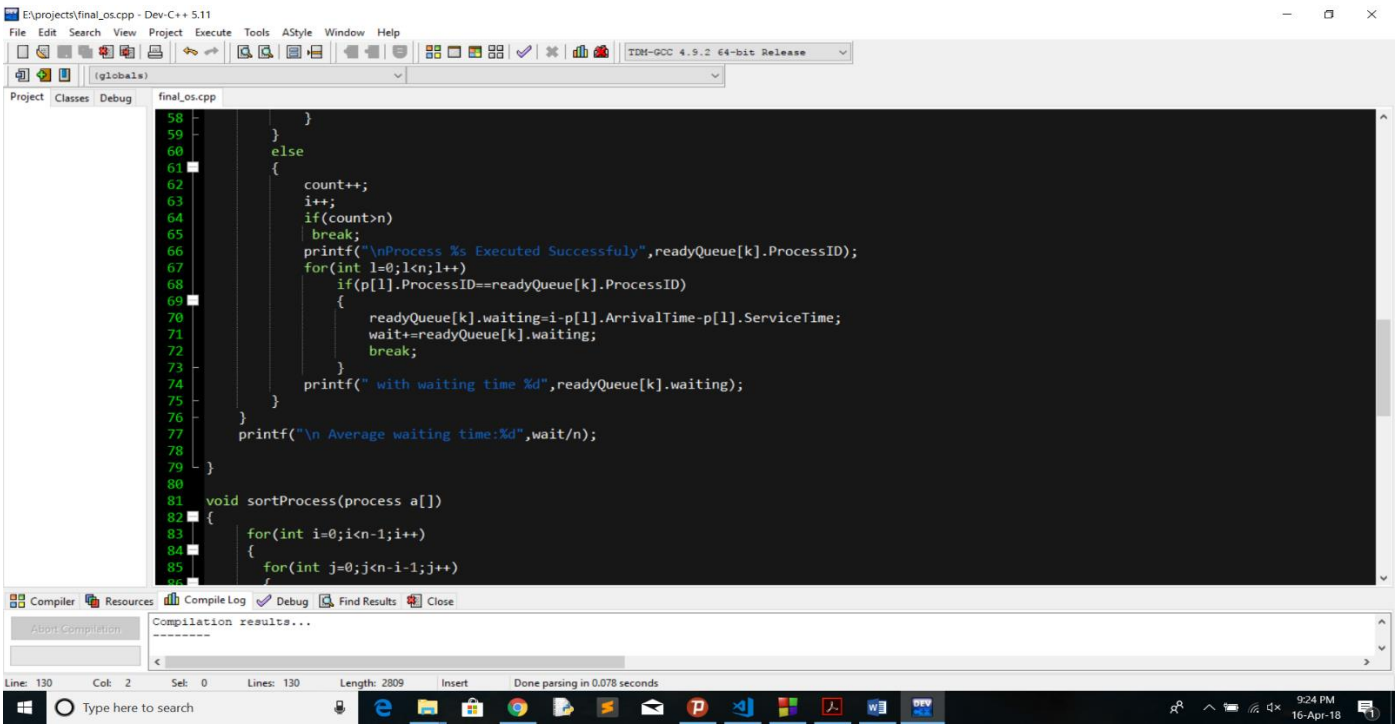



E:\projects\final_os.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
final_os.cpp
for(int i=0;i<n;i++)
{
printf("\nNow Enter Process ID:");
fflush(stdin);
scanf("%s",&p[i].ProcessID);
printf("\nNow Enter process arival time:");
fflush(stdin);
scanf("%d",&p[i].ArrivalTime);
printf("\nNow Enter process burst/service Time:");
fflush(stdin);
scanf("%d",&p[i].ServiceTime);
p[i].priority=0;
p[i].processedFlag=0;
}
sortProcess(p);
processor(p);
}
int main()
{
printf("*****************Welcome To Process Scheduler*****************\a\n");
printf("\n\n*****************Dynamic Priority Scheduling*****************\a\n");
getch();
processCreation();
}
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...

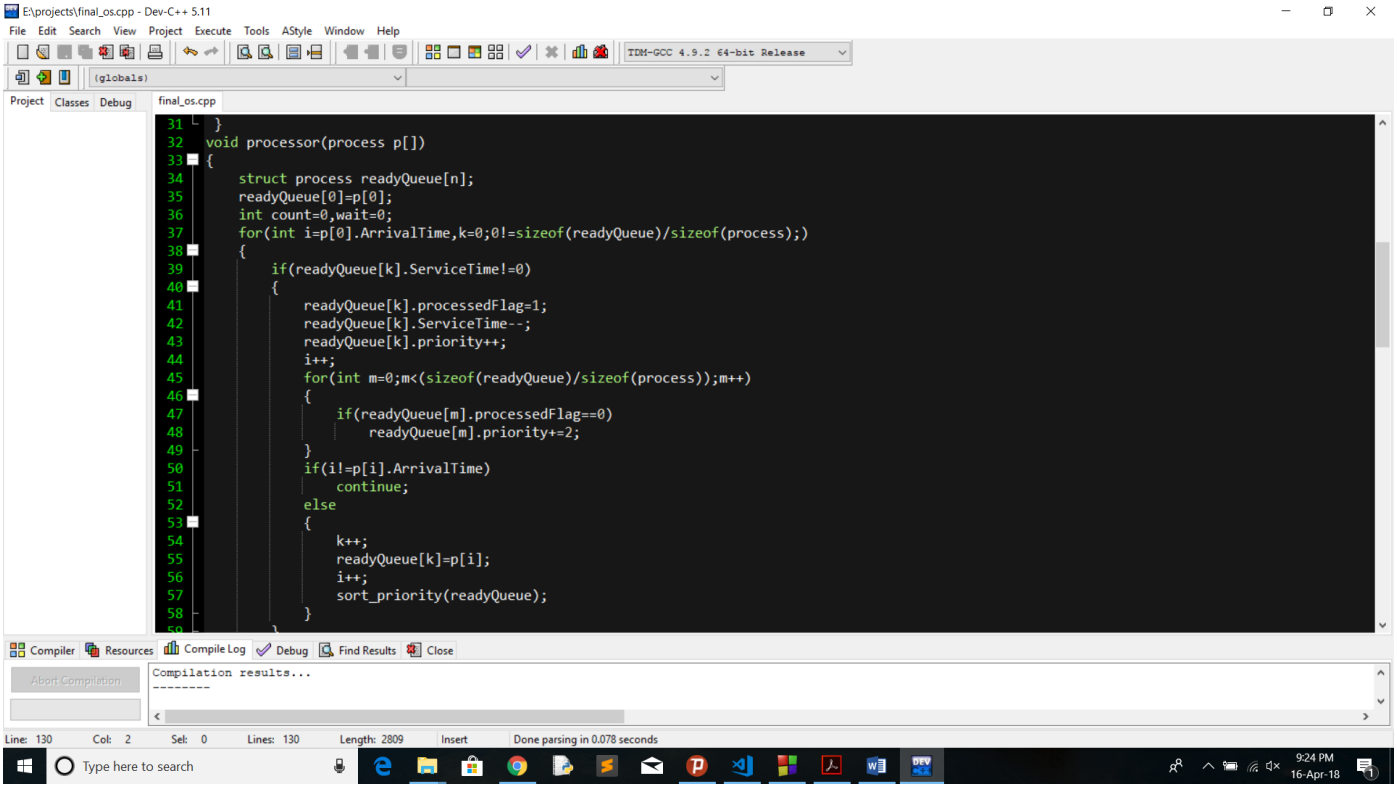
Line: 117 Col: 5 Sel: 0 Lines: 130 Length: 2809 Insert Done parsing in 0.078 seconds
Type here to search
9:25 PM
16-Apr-18


E:\projects\final_os.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
final_os.cpp
}
void sortProcess(process a[])
{
for(int i=0;i<n-1;i++)
{
for(int j=0;j<n-i-1;j++)
{
if(a[j].ArrivalTime>a[j+1].ArrivalTime)
{
process temp=a[j];
a[j]=a[j+1];
a[j+1]=temp;
}
}
}
}
void processCreation()
{
printf("Enter Number of process:");
scanf("%d",&n);
process p[n];
for(int i=0;i<n;i++)
{
printf("\nNow Enter Process ID:");
fflush(stdin);
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...

Line: 130 Col: 2 Sel: 0 Lines: 130 Length: 2809 Insert Done parsing in 0.078 seconds
Type here to search
9:24 PM
16-Apr-18


E:\projects\final_os.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
final_os.cpp
58 }
59 }
60 else
61 {
62 count++;
63 i++;
64 if(count>n)
65 break;
66 printf("\nProcess %s Executed Successfuly",readyQueue[k].ProcessID);
67 for(int l=0;l<n;l++)
68 if(p[l].ProcessID==readyQueue[k].ProcessID)
69 {
70 readyQueue[k].waiting=i-p[l].ArrivalTime-p[l].ServiceTime;
71 wait+=readyQueue[k].waiting;
72 break;
73 }
74 printf(" with waiting time %d",readyQueue[k].waiting);
75 }
76 }
77 printf("\n Average waiting time:%d",wait/n);
78
79 }
80
81 void sortProcess(process a[])
82 {
83 for(int i=0;i<n-1;i++)
84 {
85 for(int j=0;j<n-i-1;j++)
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...

Line: 130 Col: 2 Sel: 0 Lines: 130 Length: 2809 Insert Done parsing in 0.078 seconds
Type here to search
9:24 PM
16-Apr-18


E:\projects\final_os.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
final_os.cpp
31 }
32 void processor(process p[])
33 {
34 struct process readyQueue[n];
35 readyQueue[0]=p[0];
36 int count=0,wait=0;
37 for(int i=p[0].ArrivalTime,k=0;0!=sizeof(readyQueue)/sizeof(process);)
38 {
39 if(readyQueue[k].ServiceTime!=0)
40 {
41 readyQueue[k].processedFlag=1;
42 readyQueue[k].ServiceTime--;
43 readyQueue[k].priority++;
44 i++;
45 for(int m=0;m<(sizeof(readyQueue)/sizeof(process));m++)
46 {
47 if(readyQueue[m].processedFlag==0)
48 readyQueue[m].priority+=2;
49 }
50 if(i!=p[i].ArrivalTime)
51 continue;
52 else
53 {
54 k++;
55 readyQueue[k]=p[i];
56 i++;
57 sort_priority(readyQueue);
58 }
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...

Line: 130 Col: 2 Sel: 0 Lines: 130 Length: 2809 Insert Done parsing in 0.078 seconds
Type here to search
9:24 PM
16-Apr-18

Test Cases:

Process ID	Arrival Time	Service Time
P1	0	4
P2	1	1
P3	2	2
P4	3	1

Set initial priority of all the process to 0.
Output:

Average Waiting : 3.5

Boundary Conditions :

1. Maximum of 10 processes can be scheduled using this code.
2. Minimum 1 process should be there.
3. Burst time must be greater than zero.

GitHub Link: https://github.com/Deepudeepak76/K1653_B41_DEEPUDEEPAK