4B OS Lab-04 Assignment

CS182019 | Muhammad Danish

CODE SCREENSHOTS:

cs182019_Lab04.cpp

```cpp
#include<iostream>
#include <iomanip>
#include <stdlib.h>
#include <unistd.h>
#include<queue>

using namespace std;

struct Block
{
    string processState;
    int processNum;
    string schedulingInfo;
    string IOstatusInfo;
    int arrivalTime;
    int burstLength;
    int allocationTime;
    Block *programCounter;
    Block *next;

    Block(string ps, int pn, string si, string io,int at, int bt, int allt)
    {
        this->processState = ps;
        this->processNum = pn;
        this->schedulingInfo = si;
        this->IOstatusInfo = io;
        this->arrivalTime = at;
        this->burstLength = bt;
        this->allocationTime = allt;
        this->next = NULL;
    }
    ~Block(){}
};
```

Continued……

```
35  struct headNode
36  {
37      int count;
38      Block *front;
39      Block *rear;
40      headNode():front(NULL), rear(NULL){}
41  };
42
43  class Queue
44  {
45  private:
46      headNode *head = NULL;
47      int c;
48
49  public:
50      void Enqueue(string ps, int pn, string si, string io,int at, int bt, int allt);
51      void Dequeue();
52      void Scheduler(int at , int bt, int allt);
53      void Display();
54  };
55
```

cs182019_Lab04.cpp

```
55  int termination = 0;
56  int prog_counter = 98;
57  Queue q;
58  void Queue::Scheduler (int at,int bt ,int allt)
59  {
60      if(at==0)
61      {
62          termination = allt+bt;
63          prog_counter=prog_counter+2;
64          q.Enqueue("Executing",prog_counter, "priority : 1", "printer", at, bt, allt);
65          cout << endl;
66          cout << "wait for 1st process execution in : " << bt  << " seconds ." <<endl;
67          sleep(bt);
68          cout << "Process " << prog_counter<< " Terminated at " << termination << endl;
69          cout<< "Allocation Time : " << allt <<endl;
70          cout << "Burst length : " <<bt <<endl;
71          cout << "Total execution Time : " << termination << endl;
72          cout << "Process " <<prog_counter << " Exited."<<endl;
73          cout << endl;
74
75      }
76      if(at == termination)
77      {
78          termination = termination+bt;
79          prog_counter=prog_counter+2;
80          q.Enqueue("Executing",prog_counter, "priority=2", "monitor",  at, bt ,  allt);
81          cout << endl;
82          cout << "wait for more : " << bt  << " seconds ."<<endl;
83          sleep (bt);
84          cout << "Process " << prog_counter<< " Terminated at " << termination << endl;
85          cout<< "Allocation Time : " << allt <<endl;
86          cout << "Burst length : " <<bt <<endl;
87          cout << "Total execution Time : " << termination << endl;
88          cout << "Process " <<prog_counter << " Exited."<<endl;
89          cout << endl;
90      }
91
92  }
```

[*] cs182019_Lab04.cpp

```
94  void Queue::Enqueue (string ps, int pn, string si, string io,int at, int bt, int allt)
95  {
96      Block *temp = new Block(ps,pn,si,io,at,bt,allt);
97
98      if(head == NULL)
99      {
100             headNode *htemp = new headNode();
101             htemp->front = temp;
102             htemp->rear = temp;
103             htemp->count = c++;
104             head = htemp;
105     }
106
107     else
108     {
109         head->count = c++;
110         head->rear->next = temp;
111         head->rear = temp;
112     }
113 }
114 void Queue::Dequeue()
115 {
116     Block *temp = NULL;
117     temp = head->front;
118     Block *savePrev;
119
120      if(head->front == NULL)
121     {
122         cout <<"Queue underflow";
123     }
124     else
125     {
126         head->count = c--;
127         head->front = head->front->next;
128         savePrev = temp;
129         delete temp;
130     }
131 }
```

Continued……

```
133  void Queue::Display()
134  {
135      int i=0;
136      Block *temp=NULL;
137      if(head == NULL)
138      {
139          cout << "No block in list";
140      }
141      else
142      {
143          temp = head->front;
144          while(temp !=NULL)
145          {
146              cout<<endl;
147              cout << "------ Block " <<i<< "--------"<< endl;
148              cout << "Process State: " << temp->processState<< endl;
149              cout << "Process Number: " << temp->processNum<< endl;
150              cout << "Scheduling Information: " << temp->schedulingInfo<< endl;
151              cout << "IO Information: " << temp->IOstatusInfo<< endl;
152              cout << "Arrival Time: " << temp->arrivalTime<<endl;
153              cout << "Burst Length: " << temp->burstLength<<endl;
154              cout << "Allocation Time: " << temp->allocationTime << endl;
155              temp = temp->next;
156              i++;
157          }
158      }
159  }
160
```

```
160
161  int main()
162  {
163      // considering a non primitive FCFS scheduler. all process will wait until the termination of previous process.
164      q.Scheduler(0,3,0);
165      q.Scheduler(3,4,3);
166      q.Scheduler(7,5,7);
167      q.Scheduler(12,3,12);
168
169      cout << "Total execution time of all process combined : " << termination << endl;
170      cout << endl <<"------All PROCESS INFORMATION----" << endl;
171      q.Display();
172
173      q.Dequeue();
174      q.Dequeue();
175      q.Dequeue();
176      q.Dequeue();
177
178      return 0;
179  }
180
181
182
```

Continued……

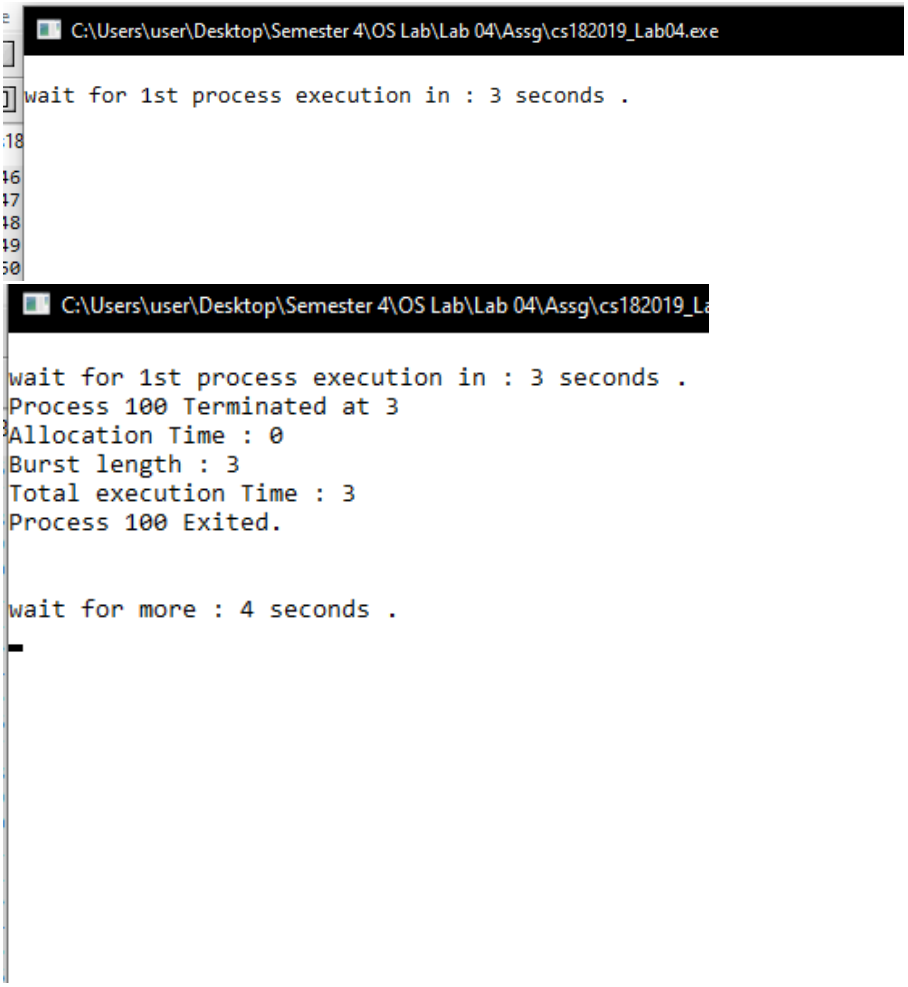
CODE OUTPUT SCREENSHOTS:

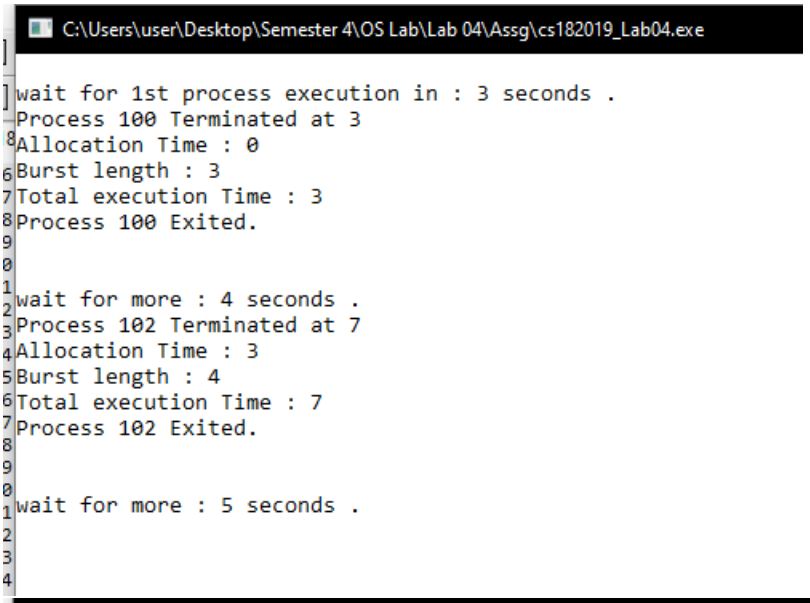
Considering a non-primitive FCFS Scheduler, next program will wait until the execution of the previous program. When the process will Enqueue than Scheduler Function will wait(for seconds equal to burst time) for its execution.

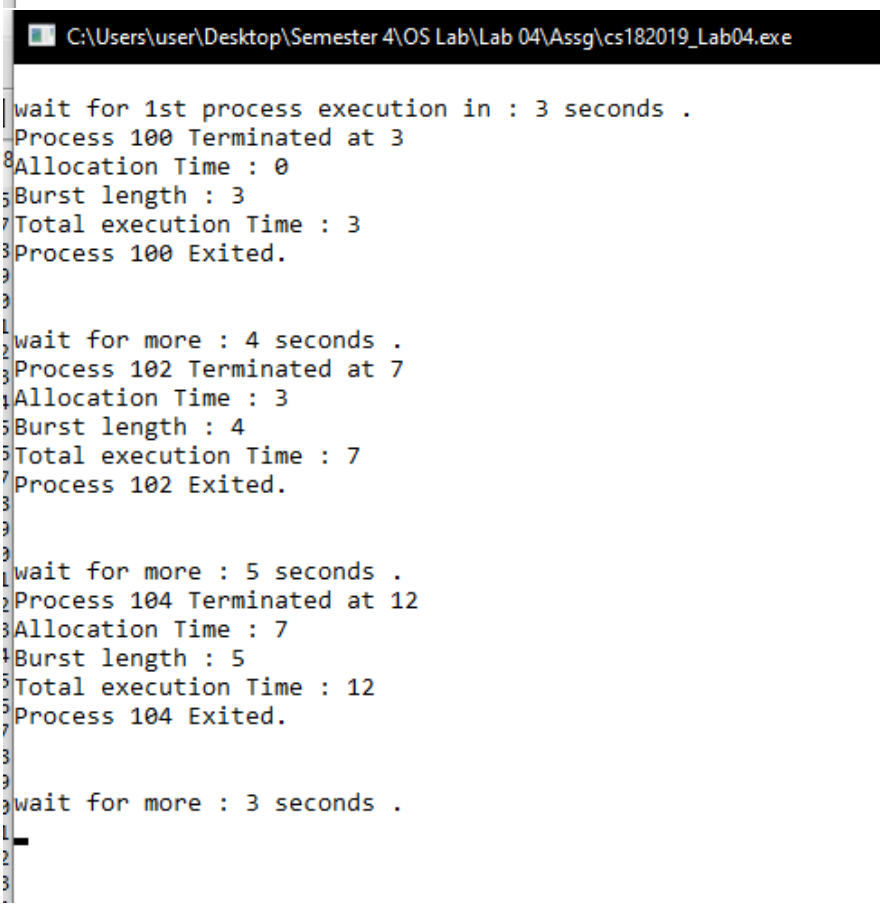
Total execution time of all processes are calculated in between and printed at Last.

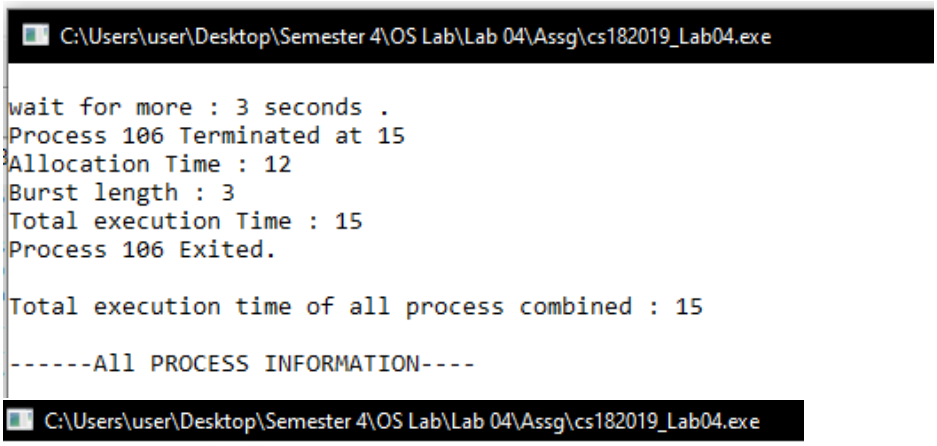
Functions: Sleep(till_burst_time) from system library is use to halt the program till the burst time of the process. Enqueing(Scheduling) of Processes is done inside Scheduler Function.

After total completion the block is Dequeue.




C:\Users\user\Desktop\Semester 4\OS Lab\Lab 04\Assg\cs182019_Lab04.exe
wait for 1st process execution in : 3 seconds .
Process 100 Terminated at 3
Allocation Time : 0
Burst length : 3
Total execution Time : 3
Process 100 Exited.
wait for more : 4 seconds .
Process 102 Terminated at 7
Allocation Time : 3
Burst length : 4
Total execution Time : 7
Process 102 Exited.
wait for more : 5 seconds .


C:\Users\user\Desktop\Semester 4\OS Lab\Lab 04\Assg\cs182019_Lab04.exe
wait for 1st process execution in : 3 seconds .
Process 100 Terminated at 3
Allocation Time : 0
Burst length : 3
Total execution Time : 3
Process 100 Exited.
wait for more : 4 seconds .
Process 102 Terminated at 7
Allocation Time : 3
Burst length : 4
Total execution Time : 7
Process 102 Exited.
wait for more : 5 seconds .
Process 104 Terminated at 12
Allocation Time : 7
Burst length : 5
Total execution Time : 12
Process 104 Exited.
wait for more : 3 seconds .




```
C:\Users\user\Desktop\Semester 4\OS Lab\Lab 04\Assg\cs182019_Lab04.exe
Process 106 Exited.

Total execution time of all process combined : 15

------All PROCESS INFORMATION----

------ Block 0--------
Process State: Executing
Process Number: 100
Scheduling Information: priority : 1
IO Information: printer
Arrival Time: 0
Burst Length: 3
Allocation Time: 0

------ Block 1--------
Process State: Executing
Process Number: 102
Scheduling Information: priority=2
IO Information: monitor
Arrival Time: 3
Burst Length: 4
Allocation Time: 3

------ Block 2--------
Process State: Executing
Process Number: 104
Scheduling Information: priority=2
IO Information: monitor
Arrival Time: 7
Burst Length: 5
Allocation Time: 7

------ Block 3--------
Process State: Executing
Process Number: 106
Scheduling Information: priority=2
IO Information: monitor
Arrival Time: 12
Burst Length: 3
Allocation Time: 12

--------------------------------
Process exited after 15.3 seconds with return value 0
```