

{CASE STUDY}

[OPERATING SYSTEM (CSE316)]

****

**STUDENT NAME :SAMA PRANAY SAI GANESH**

**sTUDENT ID:11802220.**

**GITHUB LINK &EMAIL:-samaganeshsamagaanesh@gmail.com**

**SUBMITTED TO :-VASUDHA MADAM.**

**CONTEXT**

1. **introduction**
2. **description**
3. **algorithm**
4. **code screenshots**
5. **complexity**
6. **TEST CASES**
7. **github repository**
8. **conclusion**

**INTRODUCTION**

My case study is based on the round robin algorithm but it is twisted into two Quantums i.e first quantum is 6 units of time and second quantum is 10 units of time. According my case study I have to consider the process that is completed in first quantum (6 units of time) then the second quantum(10 units of time) is applied on remaining processes and by using round robin algorithm I have to calculate waiting time, turnaround time and also I calculate average turnaround time , average waiting time. So I should explain round robin algorithm in this case study.

**DESCRIPTION**

**Round Robin Schedular** is preemptive scheduling in which every process get executed in a cyclic way i.e in round robin a particular time slice is allotted to each process which known as quantum. Every process ,which is present in the queue for processing , CPU is assigned to that process for that time quantum . Now if the execution of the process gets completed in that quantum ,then the process will get terminate otherwise the process will again go to the ready queue , and the previous process will wait for the turn to complete its execution.

* It is simple , easy to implement ,and starvation -free as all processes get fair share of cpu.
* One of the most commonly used technique in CPU scheduling as a core .
* It is preemptive as processes are assigned CPU only for a fixed slice of time at most .
* The disadvantage of it is more overhead to context switching .

**TIMES IN ROUND ROBIN**

* Completion Time :Time at which process completes its execution.
* Turn Around Time :Time difference between completion time and arrival time i.e Completion time -Arrival time.
* Waiting Time :Difference between turn around time and burst time i.e Turn-Around Time – Burst Time.

**ADVANTAGES**

* There is fairness since every process gets equal share of CPU.
* While performing a round-robin scheduling, a particular time quantum is allotted to different jobs.
* Each process get a chance to reschedule after a particular quantum time in this scheduling .
* The newly created process is added to end of ready queue .

**DISADVANTAGES**

* There is Low throughput.
* Gantt Chart seems to come too big (if quantum time is less for scheduling .)
* Time consuming scheduling for small quantums.
* There is Larger Waiting time and Response time.

**ALGORITHM**

**STEP1:-** we first have a queue where the process are arranged in first come first serve order.

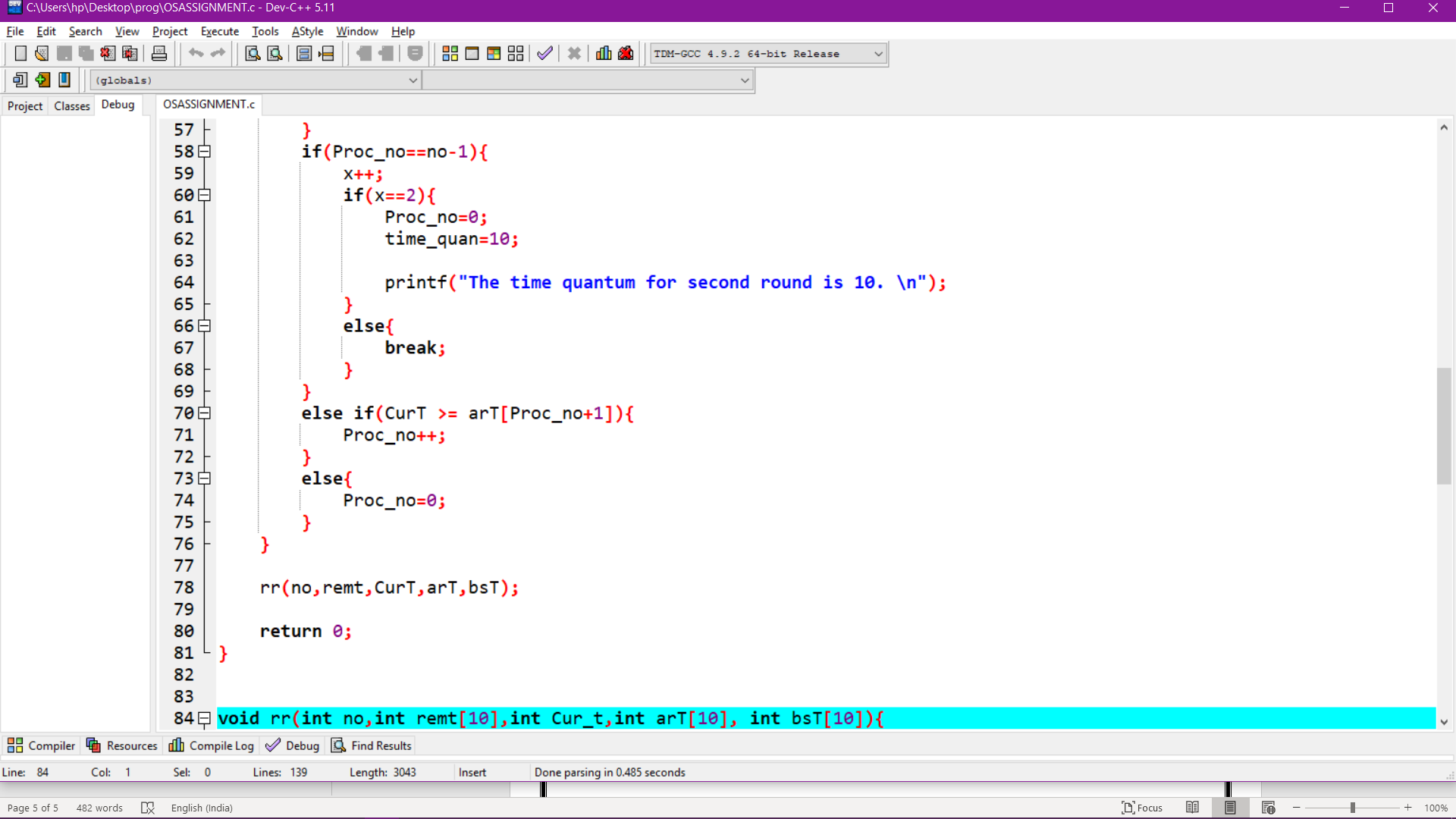
**STEP2:-** A quantum value is allotted to execute each process.

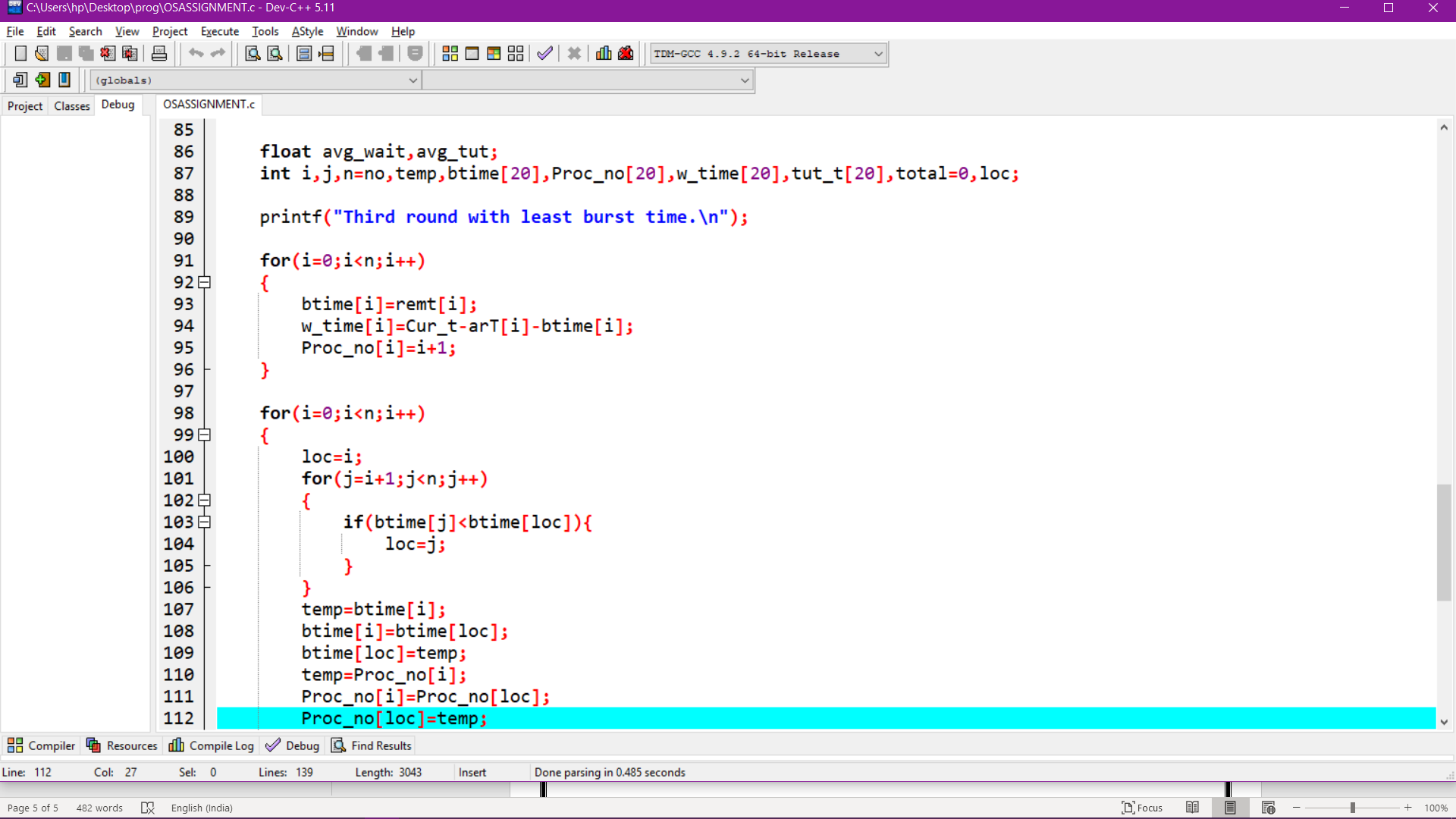
**STEP3:-** The first process is executed until the end of the quantum value . after this , an interrupt is generated and state is saved.

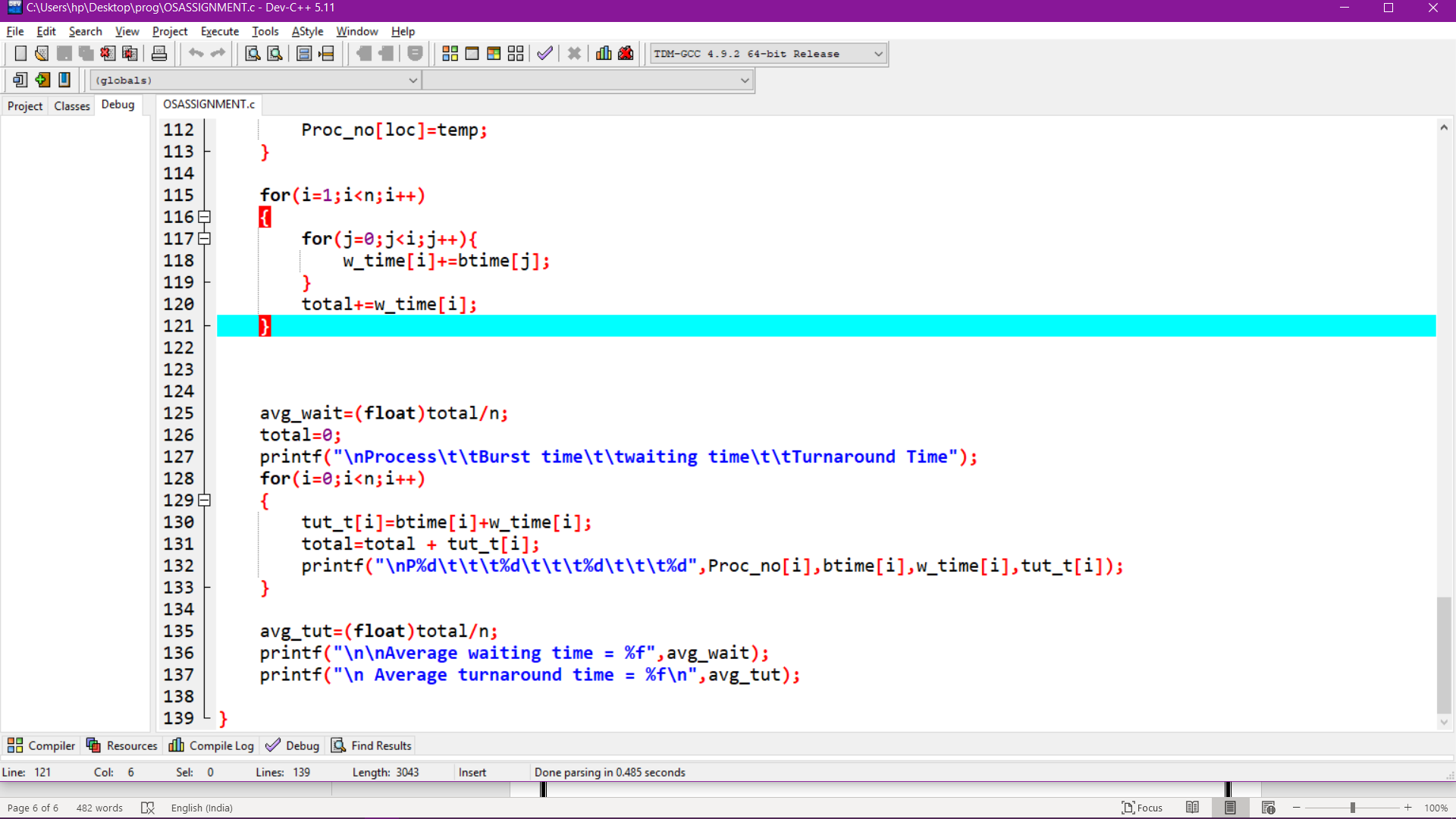
**STEP4:-**The CPU then moves to the next process and the same method is Followed.

**STEP5:-**Same steps are repeated till the processes are over.

**SCREENSHOTS OF CODE**







**COMPLEXITY**

**TEST CASES**

**PROCESS THAT ARE GIVEN IN MY CASE STUDY**

1.Time quantum=6 sec

2.Time quantum=10 sec

Note: while first quantum is executed the p1 process I.E

Waiting time =44.

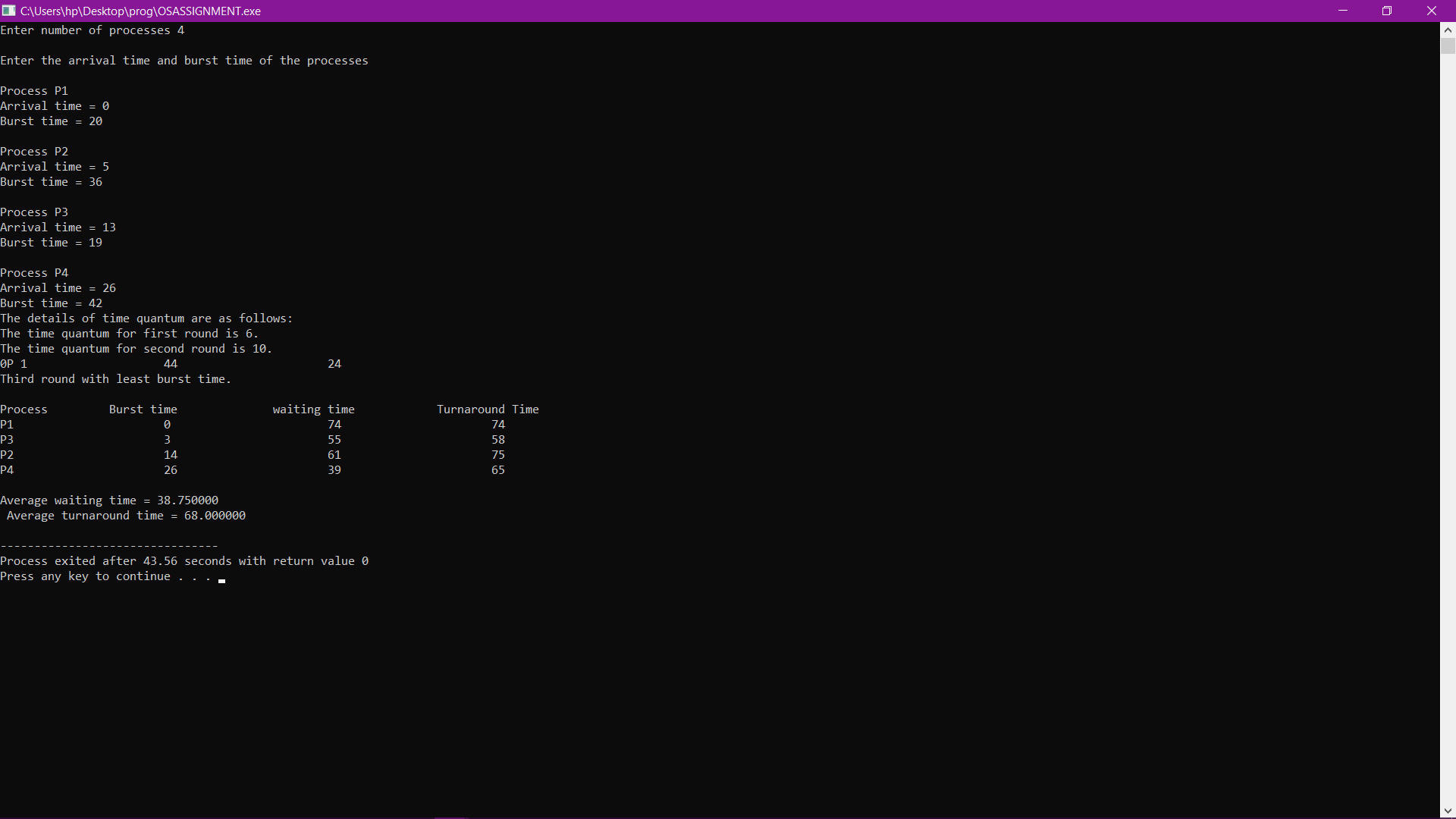
Turnaround time =24.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Processes | Arrival time | Burst time | Turnaround time | Waiting time |
| P1 | 0 | 20 | 74 | 74 |
| P2 | 5 | 36 | 55 | 58 |
| P3 | 13 | 19 | 61 | 75 |
| P4 | 26 | 42 | 39 | 65 |

Avg waiting time=38.75

Avg turnaround time=68.00

**AFTER EXECUTION:**



**GITHHUB REPOSITORY**

**CONCLUSION**

It is used to find a solution for the high turnaround time, high waiting time and the overhead of extra context switches in Simple Round Robin Algorithm. however all of them rely based on the fixed-time quantum. The proposed algorithm called **Performance analysis of Round Robin Scheduling** using adaptive approach based on smart time slice.

It approaches based on fixed time quantum in all scheduling criteria and taking into consideration the terminated and the new arrival processes. **It is recommended to use the shortest burst time concept with smart time slice** because it will give the operating system the ability to adapt to the user behavior and not vice versa. The Adaptive Round Robin Algorithm is designed to meet all scheduling criteria such as

1. maximum CPU utilization
2. maximum throughput
3. minimum average waiting time
4. minimum average turnaround time
5. minimum response time.

**THERE FOR ADAPTIVE ROUND ROBIN ALGORITHM SATISFIES SCHEDULING CRITERIA.**

**The end ….**