KUZEY KIBRIS
KAMPÜSÜ
ODTÜ
METU
NORTHERN CYPRUS
CAMPUS

CNG 334

Assignment 1 Report

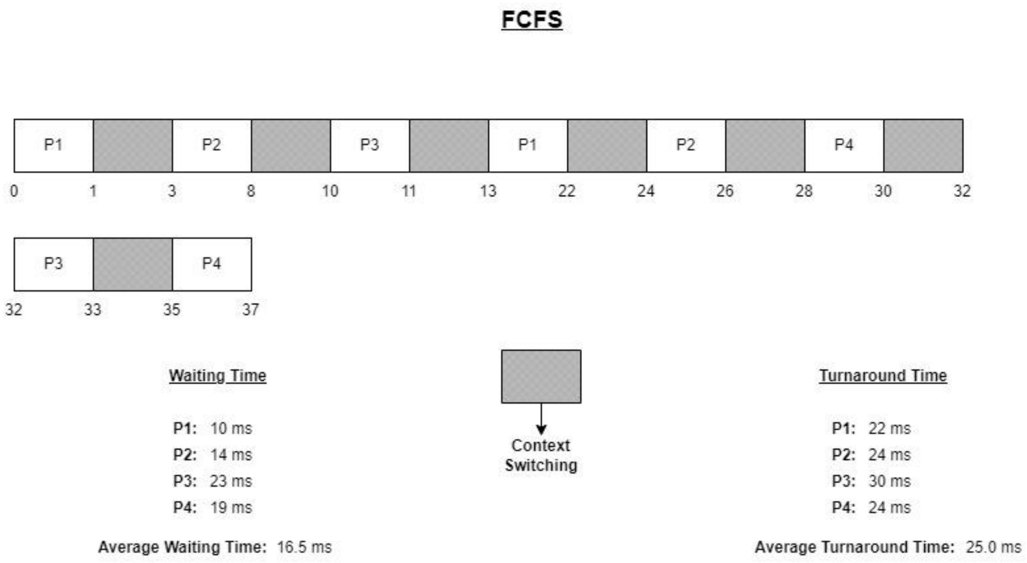
Name: Shayan

Nadeem

Student ID: 2542413

TASK 1:

1)



FCFS

Waiting Time

P1: 10 ms
P2: 14 ms
P3: 23 ms
P4: 19 ms

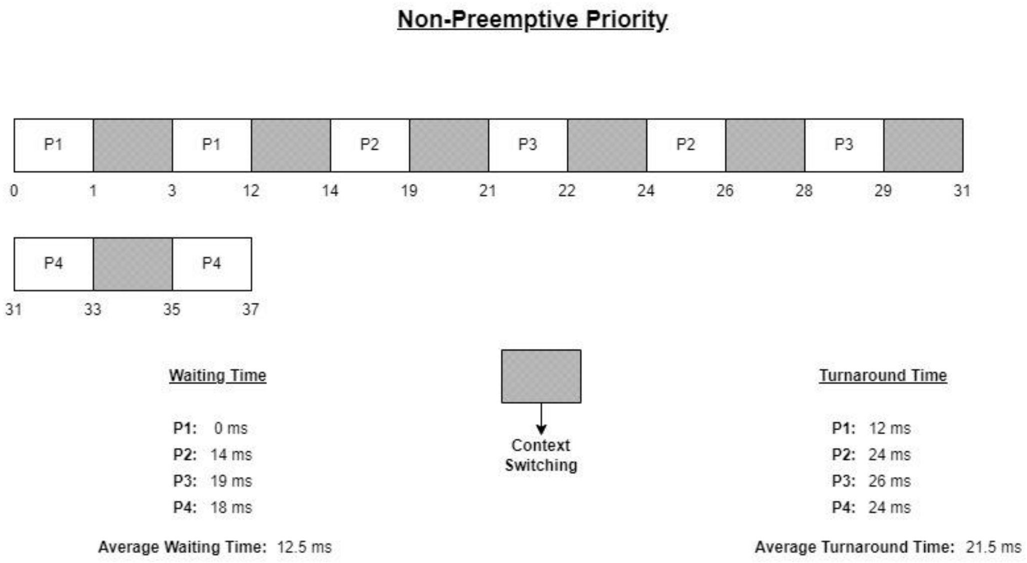
Average Waiting Time: 16.5 ms

Turnaround Time

P1: 22 ms
P2: 24 ms
P3: 30 ms
P4: 24 ms

Average Turnaround Time: 25.0 ms

2)



Non-Preemptive Priority

Waiting Time

P1: 0 ms
P2: 14 ms
P3: 19 ms
P4: 18 ms

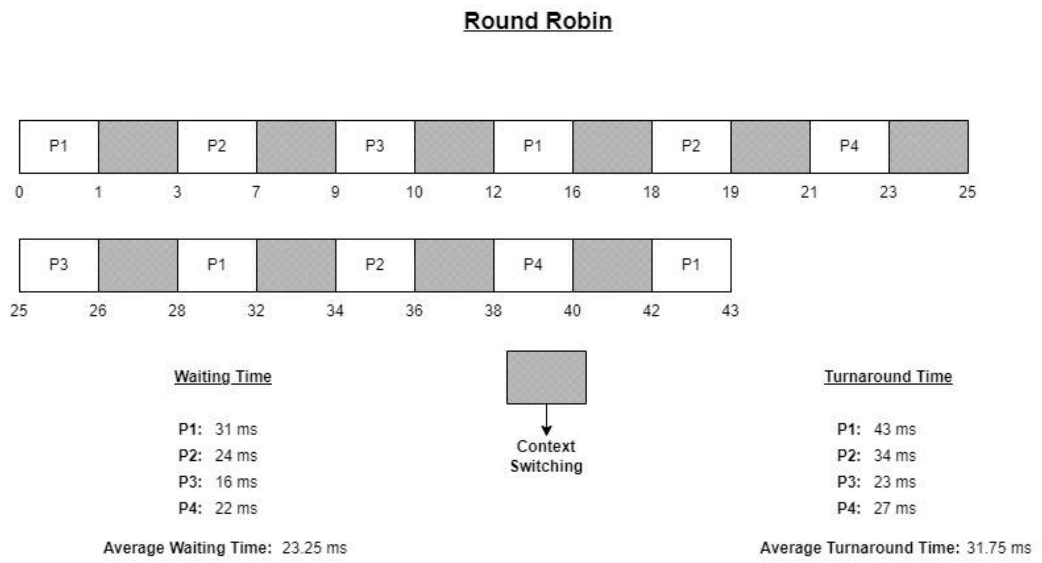
Average Waiting Time: 12.5 ms

Turnaround Time

P1: 12 ms
P2: 24 ms
P3: 26 ms
P4: 24 ms

Average Turnaround Time: 21.5 ms

3)



Round Robin

Waiting Time

P1: 31 ms
P2: 24 ms
P3: 16 ms
P4: 22 ms

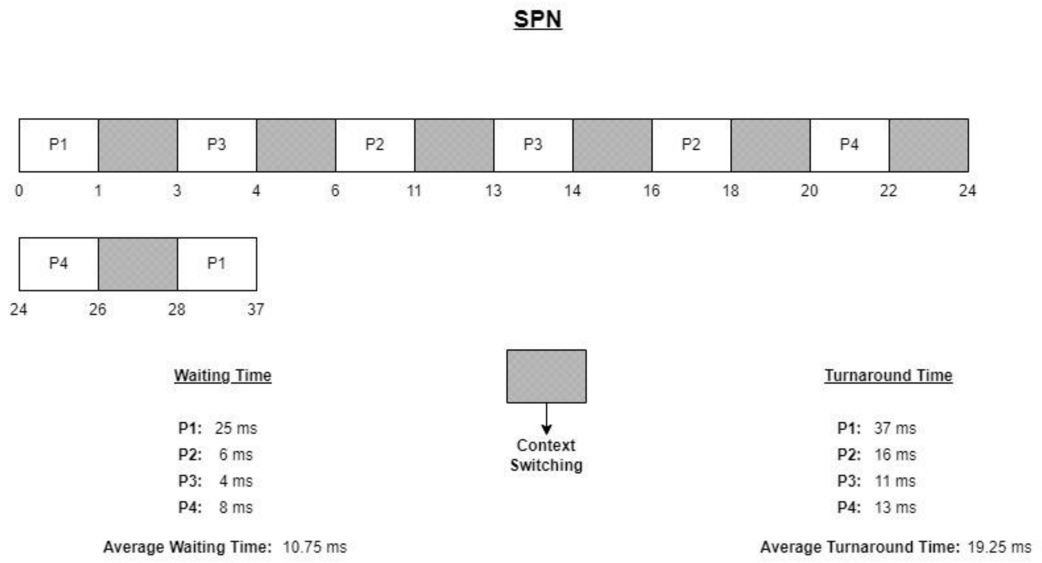
Average Waiting Time: 23.25 ms

Turnaround Time

P1: 43 ms
P2: 34 ms
P3: 23 ms
P4: 27 ms

Average Turnaround Time: 31.75 ms

4)



SPN

Waiting Time

P1: 25 ms
P2: 6 ms
P3: 4 ms
P4: 8 ms

Average Waiting Time: 10.75 ms

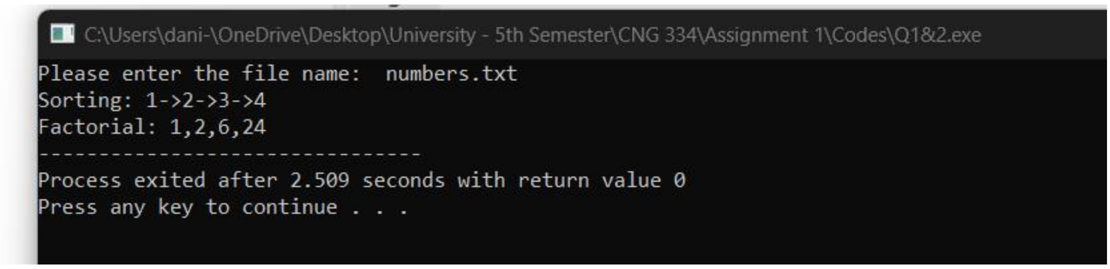
Turnaround Time

P1: 37 ms
P2: 16 ms
P3: 11 ms
P4: 13 ms

Average Turnaround Time: 19.25 ms

TASK 2:

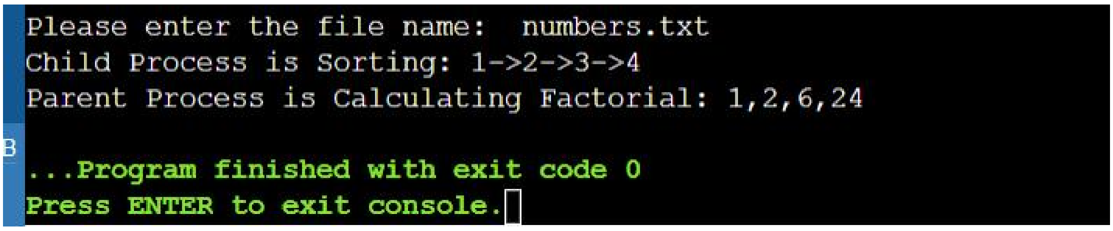
Both Functions (readSort() & readCheck()) Output Screen:



C:\Users\dani-\OneDrive\Desktop\University - 5th Semester\CNG 334\Assignment 1\Codes\Q1&2.exe

```
Please enter the file name:  numbers.txt
Sorting: 1->2->3->4
Factorial: 1,2,6,24
--------------------------------
Process exited after 2.509 seconds with return value 0
Press any key to continue . . .
```

Child And Parent Processes Output Screen:

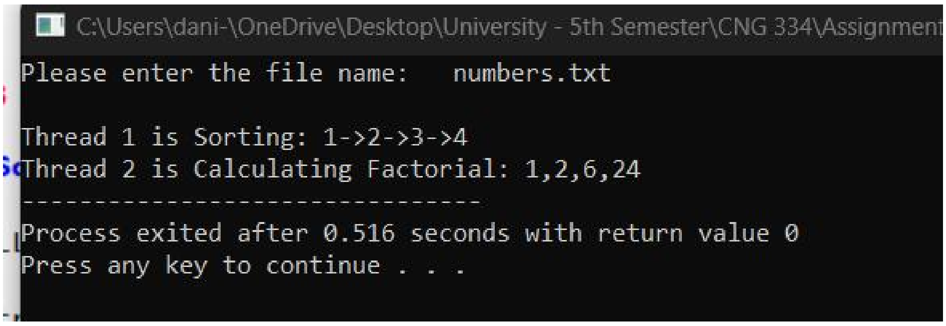


```
Please enter the file name:  numbers.txt
Child Process is Sorting: 1->2->3->4
Parent Process is Calculating Factorial: 1,2,6,24

...Program finished with exit code 0
Press ENTER to exit console.
```

//I was not able to implement this in a normal DEVC++ compiler due to library issues so I had to use an online compiler for this part only//

Threading Output Screen:

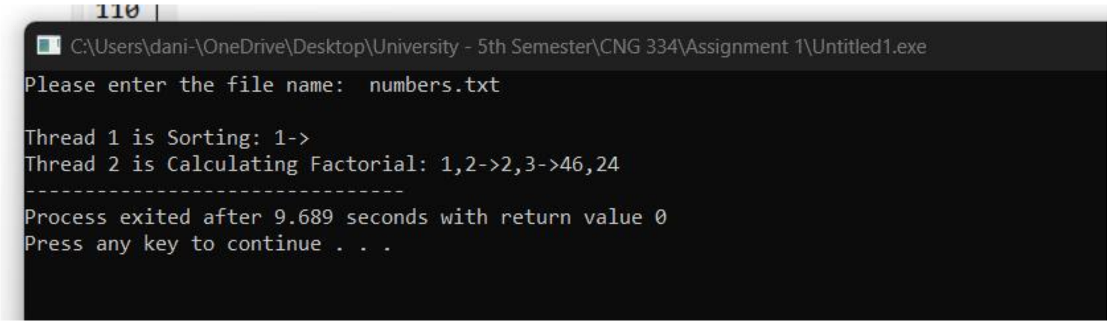


C:\Users\dani-\OneDrive\Desktop\University - 5th Semester\CNG 334\Assignment

```
Please enter the file name:   numbers.txt

Thread 1 is Sorting: 1->2->3->4
Thread 2 is Calculating Factorial: 1,2,6,24
--------------------------------
Process exited after 0.516 seconds with return value 0
Press any key to continue . . .
```

Threading (Including Waiting) Output Screen:



C:\Users\dani-\OneDrive\Desktop\University - 5th Semester\CNG 334\Assignment 1\Untitled1.exe

```
Please enter the file name:  numbers.txt

Thread 1 is Sorting: 1->
Thread 2 is Calculating Factorial: 1,2->2,3->46,24
--------------------------------
Process exited after 9.689 seconds with return value 0
Press any key to continue . . .
```

TASK 3:

1: Concurrency refers to the ability of multiple tasks to run in similar time periods, concurrently (one after another) but not simultaneously while parallelization involves the actual simultaneous execution of multiple tasks simultaneously using multiple CPU cores or threads. Multi Programming increases CPU utilization by allowing the CPU to switch between executing different tasks, which reduces idle time and increases overall efficiency. CPU is (in most cases) never allowed to stay IDLE. This will ultimately result in a better performance.

2: **Short Term Scheduling:** It is used to select processes to allocate CPU time to from an already **active (ready) queue of processes**. It maximizes the CPU utilization and minimizes the response time. Examples: SPN etc.

Medium Term Scheduling: Only goal is to retrieve processes from disk and putting them in memory or vice versa (in case of **I/O interrupts or blocking**). It is mainly used to resource optimization.

Long Term Scheduling: This is mostly done for Batch Processing Systems where **new processes** are selected to be loaded into the main memory. This type of scheduling ensures that the system is not overloaded with too many processes, and there is enough memory to accommodate them.