Internal Assignment – Practical

Paper Name – Operating Systems

B.Sc. (Hons.) Computer Science, Semester 3

Anshul Verma (19/78065)

Q1.

```
#include <sys/types.h>
#include <stdio.h>
#include <unistd.h>


#define SIZE 5


int nums[SIZE] = {0,1,2,3,4};


int main()
{
int i;
pid t pid;


    pid = fork();


    if (pid == 0) {
        for (i = 0; i < SIZE; i++) {
            nums[i] *= -i;
            printf("CHILD: %d ",nums[i]); /* LINE X */
        }
    }
    else if (pid > 0) {
        wait(NULL);
        for (i = 0; i < SIZE; i++)
            printf("PARENT: %d ",nums[i]); /* LINE Y */
    }
    return 0;
}
```

What output will be at Line X and Line Y? Justify your answer.

Sol: 1

Output:

Iteration	Line X	Line Y
1	CHILD: 0	PARENT: 0
2	CHILD: -1	PARENT: 1
3	CHILD: -4	PARENT: 2
4	CHILD: -9	PARENT: 3
5	CHILD: -16	PARENT: 4

Here, as we can see, only the data for child process has changed not the ~~par~~ data for parent process. This is because:

The changes made in child process occurs in only its copy of data and thus, it won't affect the parent process.

Q2.

The following processes are being scheduled using a preemptive, round-robin scheduling algorithm. Each process is assigned a numerical priority, with a higher number indicating a higher relative priority. In addition to the processes listed below, the system also has an ***idle task*** (which consumes no CPU resources and is identified as P_{idle}). This task has priority 0 and is scheduled whenever the system has no other available processes to run. The length of a time quantum is 10 units. If a process is preempted by a higher-priority process, the preempted process is placed at the end of the queue.

Thread	Priority	Burst	Arrival
P_1	40	20	0
P_2	30	25	25
P_3	30	25	30
P_4	35	15	60
P_5	5	10	100
P_6	10	10	105

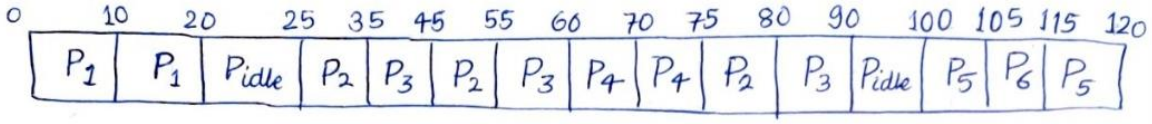
a. Show the scheduling order of the processes using a Gantt chart.

b. What is the turnaround time for each process?

c. What is the waiting time for each process?

d. What is the CPU utilization rate?

Sol:2 (a) According to given table, we can draw Gantt Chart as follows:



(b)

Thread	Arrival Time (AT)	Completion Time (CT)	Turnaround Time (TT) = (CT − AT)
P_1	0	20	20-0 = 20
P_2	25	80	80-25 = 55
P_3	50	90	90-30 = 60
P_4	60	75	75-60 = 15
P_5	100	120	120-100 = 20
P_6	105	115	115-105 = 10

(c)

Thread	Burst Time (BT)	TurnAroud Time (TT)	Waiting Time (WT) = (TT − BT)
P_1	20	20	20-20 = 0
P_2	25	55	55-25 = 30
P_3	25	60	60-25 = 35
P_4	15	15	15-15 = 0
P_5	10	20	20-10 = 10
P_6	10	10	10-10 = 0

(d) CPU Utilization rate = $\frac{105}{120}$ = 87.5 %