



## **Usman Institute of Technology**

### **Department of Computer Science Fall 2022**

Name: Muhammad Waleed

Roll no: 20B-115-SE

Course: Operating Systems (CS312)

Course Instructor: Ma'am Shabina Mushtaq

Date: 7-Jan-2023

#### Tasks:

1. Start five processes using multiprocessing.Process objects , each process will update shared memory Value object using their own target function (callable object to be invoked by the run())

Muhammad Waleed  
20b-115-se  
SE-B  
Lab#12  
Ma'am Shabina Mushtaq

method). After execution of all child processes, parent process should display the value of the object.

```
from multiprocessing import Process, Value

def update_value(val,i):
    val.value += 1
    print(f"Value after child process {i}:", val.value)

if __name__ == '__main__':
    shared_value = Value('i', 0)
    print("Value before child processes:", shared_value.value)
    processes = []
    for i in range(5):
        p = Process(target=update_value, args=(shared_value,i,))
        processes.append(p)
        p.start()
    for p in processes:
        p.join()
    print("Value after all child processes:", shared_value.value)
```

Output:

```
PS C:\Users\hp\Desktop\Lab#12> & C:/
Value before child processes: 0
Value after child process 1: 1
Value after child process 0: 2
Value after child process 2: 3
Value after child process 3: 4
Value after child process 4: 5
Value after all child processes: 5
PS C:\Users\hp\Desktop\Lab#12>
```

2. Generate 10 random numbers between 0 and 10, and calculate square of each number such that process#1 calculates square of first five numbers and process#2 calculates square of remaining five numbers, Store the square results in an array (shared memory region) using multiprocessing module.

Muhammad Waleed  
20b-115-se  
SE-B  
Lab#12  
Ma'am Shabina Mushtaq

```
from multiprocessing import Process, Array
import random

def calculate_squares(arr, start, end):
    for i in range(start, end):
        arr[i] = arr[i] ** 2

if __name__ == '__main__':
    numbers = Array('i', [random.randint(0, 10) for _ in range(10)])
    print("Original numbers:", numbers[:])
    p1 = Process(target=calculate_squares, args=(numbers, 0, 5))
    p2 = Process(target=calculate_squares, args=(numbers, 5, 10))
    p1.start()
    p2.start()
    p1.join()
    p2.join()
    print("Squared numbers:", numbers[:])
```

Output:

```
PS C:\Users\hp\Desktop\Lab#12> & C:/Users/hp/AppData/L
Original numbers: [5, 0, 10, 2, 5, 1, 5, 4, 0, 2]
Squared numbers: [25, 0, 100, 4, 25, 1, 25, 16, 0, 4]
```