

National University of Computer and Emerging Sciences, Lahore Campus



Course:	Operating Systems	Course Code:	CS 2006
Program:	BS (CS)	Semester:	Spring 2023
Due Date	7-May-2023 at 11:59 pm	Total Marks:	50 marks
Type:	Assignment 3	Page(s):	4

Important Instructions:

1. Submit the solution in a zipped file named as your roll number., i.e., 21L-1111.zip
2. First three questions are coding based so send their working codes and last two questions can be solved on MS word.
3. **You are not allowed to copy solutions from other students. We will check your code for plagiarism using plagiarism checkers. If any sort of cheating is found, heavy penalties will be given to all students involved.**
4. Late submission of your solution is not allowed

Question # 1:

[Marks:10]

Write a multithreaded program that calculates various statistical values for a list of numbers. This program will be passed a series of numbers on the command line and will then create three separate worker threads. One thread will determine the average of the numbers, the second will determine the maximum value, and the third will determine the mini-mum value.

For example, suppose your program is passed the integers

90 81 78 95 79 72 85

The program will report

The average value is 82

The minimum value is 72

The maximum value is 95

The variables representing the average, minimum, and maximum values will be stored globally. The worker threads will set these values, and the parent thread will output the values once the workers have exited.

Question # 2:

[Marks:10]

Write a multithreaded program that outputs prime numbers. This program should work as follows: The user will run the program and will enter a number on the command line. The program will then create a separate thread that outputs all the prime numbers less than or equal to the number entered by the user.

Question # 3:**[Marks:10]**

The Fibonacci sequence is the series of numbers 0, 1, 1, 2, 3, 5, 8, Formally, it can be expressed as:

fib0 = 0

fib1 = 1

fibn = fibn-1 + fibn-2

Write a multithreaded program that generates the Fibonacci sequence. This program should work as follows: On the command line, the user will enter the number of Fibonacci numbers that the program is to generate. The program will then create a separate thread that will generate the Fibonacci numbers, placing the sequence in data that can be shared by the threads (an array is probably the most convenient data structure). When the thread finishes execution, the parent thread will output the sequence generated by the child thread. Because the parent thread cannot begin outputting the Fibonacci sequence until the child thread finishes, the parent thread will have to wait for the child thread to finish.

Question # 4:**[Marks:10]**

Consider the following code for a simple Stack:

```
class Stack {
private:
int* a; // array for stack
int max; // max size of array
int top; // stack top
public:
Stack(int m) {
a = new int[m]; max = m; top = 0;
}
void push(int x) {
while (top == max); // if stack is full then wait
a[top] = x;
++top;
}
int pop() {
while (top == 0); // if stack is empty then wait
int tmp = top;
--top;
return a[tmp];
}
};
```

Assuming the functions push and pop can execute concurrently, synchronize the code using semaphores. Also, replace the busy waiting with proper waiting.

Playing Cricket

You need at least one bowler and one batsman to play cricket.

There are two threads, one for bowlers and one for batsmen. There is room for only one practice session. A practice session is possible if there is a bowler and a batsman available. If a batsman arrives, and a bowler is already available, he will engage a bowler for a practice session and vice versa. If however, a new batsman arrives and a bowler is not available, the batsman will hold the court and wait for the bowler to arrive and vice versa. No two bowlers can coexist the same time the same way as no two batsmen. Synchronize your solution by using semaphore.

//declare any shared variables or semaphore here

Batsmen	Bowlers
<p data-bbox="313 701 691 743">Critical Section: Play()</p>	<p data-bbox="933 701 1312 743">Critical Section: Play()</p>

GOOD LUCK!