# National University of Computer and Emerging Sciences

**FAST School of Computing Fall-2023 Islamabad Campus**

Serial No:

# Operating Systems Lab(CS, AI)

## 2nd Sessional Exam Total Time: 2.5 Hour Total Marks: 50

Tuesday, 14th November, 2023

_____ **Signature of Invigilator**

## Course Instructors

Javeria Zia, Faria Eman

_____ _____ _____ _____

**Student Name Roll No. Course Section Student Signature**

**DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.**

**Instructions:**

1. Attempt on computer. Attempt all of them. Read the question carefully, understand the question, and then attempt it.
2. No additional sheet will be provided for rough work. Use the back of the last page for rough work.
3. If you need more space, write on the back side of the paper and clearly mark question and part number etc.
4. After asked to commence the exam, please verify that you have **four (4)** different printed pages including this title page. There are total of **3** questions.
5. Calculator sharing is strictly prohibited.
6. Use permanent ink pens only. Any part done using soft pencil will not be marked and cannot be claimed for rechecking.

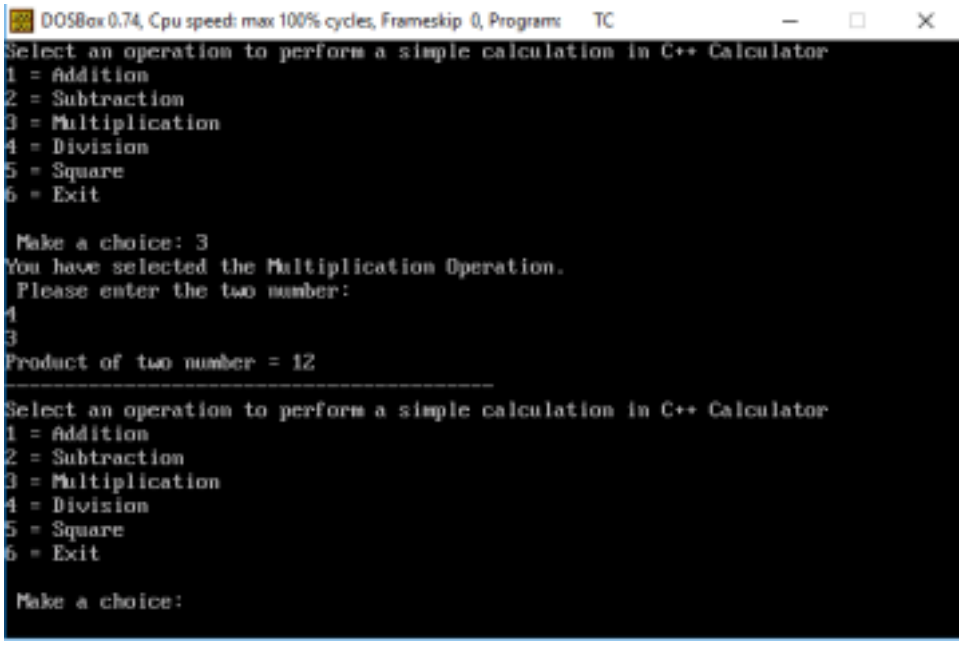
| | Q-1 | Q-2 | Q-3 | Total |
|---|---|---|---|---|
| **Marks Obtained** | | | | |
| **Total Marks** | **10** | **20** | **20** | **50** |


# National University of Computer and Emerging Sciences

**FAST School of Computing Fall-2023 Islamabad Campus**


## Question 1 [10 Marks]

Write a shell script program for calculator as shown below.




## Question 2 [20 Marks]

Write a program which take string from user and you have to do following tasks with the string.

1. String will be passed to the second **process** and that process will **Reverse** that string and return that to the main process. (hint: use pipes here)
2. Now by **using fork(), create another process** it will return the **Length of the string** to the main process without using pipes and global variables.
3. Now create **multiple threads for each character** present in the string and **Add 2** in the ASCII of each character and return values to the main process.
4. The main will **Sort all characters** in alphabetical order and display the

result. Note: Take short strings for this task just for ease.

## Question 3 [20 Marks]

Develop a program that employs multi-threading to facilitate the students aim to buy the maximum number of items with the minimum proposed balance. The challenge is to optimize their purchases while staying within budget constraints. Mutex synchronization will be crucial to ensuring fair and accurate calculations in this competitive shopping scenario.

In the main process, gather student details including their name, student ID, proposed budget, and the shopping list containing items with their respective prices. Each student will have a unique proposed budget for the shopping challenge.





Pass these details to individual threads representing each student. Each thread should simulate the student's shopping experience by iteratively selecting items from the shopping list based on their prices and the remaining budget. Implement mutex synchronization to avoid conflicts when updating the total

cost and ensuring that each student's balance is accurately maintained.

Implement two scenarios: one with mutex implementation to demonstrate how it resolves concurrent access issues, and another without mutex to showcase potential problems such as incorrect total costs and budget overruns due to simultaneous updates. This comparative presentation will highlight the significance of mutexes in maintaining fairness and accuracy in a competitive shopping simulation where students aim to maximize their purchases with a limited budget.