

Reg. No.

CSE1 Group-1 Subgroup-A Odd



Indian Institute of Information Technology, Surat
Department of Computer Science & Engineering
End Semester Lab Examination, 1st Sem, 2025-26
Fundamentals of Computer and C Programming (CS101)

Timing: 11:00 AM to 1:00 PM

Date: Nov 17, 2025

Marks: 25

Instructions:

- Work on the computer assigned to you. You should log-in with the **exam** user whose details are given on the board.
- You will have **45 minutes** to code the solutions to both the problems.
- After 45 minutes, you must stop coding. **Do not modify** your code after this time.
- The remaining time will be used for a **viva** (oral examination). Lab instructors will evaluate your code and ask questions about your solution and related concepts.
- You will be graded on:
 - **Correctness:** Does the program work as specified? Does it handle all the edge cases? Code that does not compile at the end of 45 minutes will receive a **zero** for the 'correctness' portion.
 - **Adherence to instructions:** Does the program follow all instructions, such as writing functions with specified function prototypes or using specified structures? Does the program print appropriate user instructions and error messages?
 - **Best practices:** Code clarity, proper indentation, use of comments, and variable naming.
- Since we do not have an uninterrupted power supply, please save your work frequently.
- You may (and should!) carry pen and blank paper with you, but no other devices including phones, smartwatches, and calculators are allowed.
- You must write your solutions in C. You may include any library that is present on your system. Lab instructors will not debug or help compile your code.

1. Make Lowercase

(10)

Write a C program that asks a user to enter a single character. Write a function `void make_lowercase(char *c)`; that takes a single scanned character by reference and modifies it without returning anything. It then prints the updated character.

- If the character is uppercase alphabet, convert it to the lowercase
- Otherwise (even if it is not an alphabet character), do not change it.

E.g.: On input **C** you should output **c** and on input **?** you should output **?**.

2. Min Absolute Sum

(15)

Write a C program that:

- Asks the user to enter a number, $2 \leq n \leq 1000$. Report error if n does not fall in this range and terminate the program.
- Reads n integers from the user. Report error and terminate the program if any of the numbers has absolute value strictly more than $1,00,00,000$.
- Finds a min-abs-sum pair (x, y) from the input that minimizes $|x+y|$. You may break ties arbitrarily. The order of the numbers you print in the pair does not matter.

E.g. on input $n=4$ and numbers $3, -3, 1, 9$, you should output $(-3, 3)$.

Explanation: The pair $(-3, 3)$ has an absolute sum of $|-3+3|=0$. It can be verified that no other pair from the input (e.g., $(-3, 1)$ or $(3, 1)$) has a smaller absolute sum.