Programming in C - Week 4

Dated: 25-04-26

Attempt any all questions.

Wiping your tears with the question paper is not allowed!

Question 1: Write a program that

- Declares an array of 3 integers.
- Takes 3 inputs from the user.
- Prints the entered numbers using a function print_arr().

Question 2: Write a program using rand() to:

- Fill an array of 10 integers with values from 1 to 100. *(remember rand()%100 is just going to get numbers from 0-99 we need 1 to 100 which can be achieved by adding 1 to the expression.)*
- Find and print the minimum value in the array.
- ! Use time seed to get true random numbers !

Question 3: Implement a theatre seat reservation system:

- 10 seats (array of 10 elements initialized to 0).
- Ask the user if they want to reserve (y/n).
- If yes, show the seat map and ask which seat (1–10).
- If the seat is available, reserve it (set to 1). If not, show an error.
- Repeat until user inputs 'n'.

Question 4: Write a program that:

- Takes two 2D arrays (3×3) as input (hardcoded or via user).
- Adds them element-wise into a third matrix.
- Prints the result.

Question 5: Write a program that:

- Takes a 2D array (e.g., 4×4) as input.
- Ask the user for a number.
- Searches for the number and prints its row and column index (if found).

  Example output: Enter number to search: 42

  Found at position [2][3] !

Question 6: Using binary search to search a given sorted array of 10 student roll numbers in ascending order.
Write a program that:

- Takes a roll number from the user.
- Uses **binary search** to check if that roll number exists.
- If found, print the index and a success message.
- If not, print “Roll number not found.”

Constraints:

- Use int roll_numbers[10] = {101, 104, 107, 110, 115, 120, 123, 130, 135, 140};
- Do not use linear search — use binary search logic.

Question 7: You are given an array of 8 product prices.
Write a program that:

- Sorts the prices using **selection sort**.
- Prints the prices before and after sorting.

Constraints:

- Use float prices[8] = {45.50, 12.99, 89.00, 24.75, 66.10, 10.00, 32.40, 78.90};
- Apply selection sort manually (no built-in sort).
- After each outer loop pass, print the partially sorted array.
- At the end print the sorted array.