

CSE117 Problem Solving with C

Handout - Lab Session - 10
Strings & Structures

Objective:

- To write programs that read, write and manipulate strings.
- To write programs that use array of strings.
- To write programs that use the string functions in the string library.
- To use type definition statements in programs.
- To declare and use structures in programs.

Pre-Lab: Go through the concepts of strings, structures and unions.

During Lab: Solve all the exercise problems. You should work on the additional set of programs only after completing this week's tasks.

Read Chapters 11 & 12

Lab Exercises

Exercise 1: String Length

Write a program to find the length of the string.

(Rewrite the program using builtin string function strlen())

Sample Test Cases	Input	Output
Test Case 1	Enter the string : Hello World !	Length : 13
Test Case 2	Enter the string : check123	Length : 8

Exercise 2: String Concatenation

Write a program to concatenate two strings with a space.

(Rewrite the program using builtin string function strcat())

Sample Test Cases	Input	Output
Test Case 1	Enter first string : Hello Enter second string : World	Concatenated String : Hello World
Test Case 2	Enter first string : CSM Enter second string :CSD	Concatenated String : CSM CSD

Exercise 3: String Copy

Write a program to copy one string to another.

(Rewrite the program using builtin string function strcpy())

Sample Test Cases	Input	Output
Test Case 1	Enter string : CSM	Copied String is CSM
Test Case 2	Enter string : Hello ANITS	Copied String is Hello ANITS

Exercise 4: String Comparison

Write a program to compare two strings.

(Rewrite the program using builtin string function strcmp())

Sample Test Cases	Input	Output
Test Case 1	Enter first string : Hello Enter second string : Hello	Strings are equal.
Test Case 2	Enter first string : Hello Enter second string : ANITS	Strings are not same.

Exercise 5: String Extraction

Write a program that extracts part of the given string from the specified position.

(extract *p* characters from position *n*.)

Sample Test Cases	Input	Output
Test Case 1	Enter the string : Working with strings is fun Enter the poition : 4 Enter the number of characters : 4	Extracted string: king
Test Case 2	Enter the string : Working with strings is fun Enter the poition : 4 Enter the number of characters : 0	No string extracted

Exercise 6: Case Conversion

Write a program that converts all lowercase characters in a given string to its equivalent uppercase character.

Sample Test Cases	Input	Output
Test Case 1	Enter the string : hello world	HELLO WORLD
Test Case 2	Enter the string : strings123	STRINGS123

Exercise 7: Student Database

Create a structure to specify data on students given below:

Roll number, Name, Department, Year of joining

Assume that there are not more than 200 students in the department.

- Write a function to print names of all students who joined in a particular year.
- Write a function to print the data of a student whose roll number is given.

Sample Test Cases	Input & output
Test Case 1	<p>Enter the number of students : 3</p> <p>Enter the details of students 1:</p> <p>1 Arjun CSM 2022</p> <p>Enter the details of student 2:</p> <p>2 Arpita CSD 2022</p> <p>Enter the details of student 3:</p> <p>3 Asritha CSM 2022</p> <p>Enter year : 2022</p> <p>Names of the students admitted in the year 2022:</p> <p>Arjun Arpita Asritha</p> <p>Enter the student's roll number : 2</p> <p>Details of the student: Roll Number : 2 Name: Arpita Department : CSD Year of joining: 2022</p>

Exercise 8: Bank Database

Create a structure to specify data of customers in a bank. The data to be stored is: Account number, Name, Account balance. Assume that there are not more than 200 customers in the bank.

The program should allow the user to perform the following operations;

- a. Add a new account to the database
- b. Display the details of an account given its account number.
- c. Withdraw money from an account given its account number.
- d. Deposit money into an account given its account number.
- e. Exit the program.

Sample Test Cases	Input
Test Case 1	<div>1. Add account</div> <div>2. Display account details</div> <div>3. Withdraw money</div> <div>4. Deposit money</div> <div>5. Exit</div> <div>Enter Choice: 1</div> <div>Enter account number: 1001</div> <div>Enter name: Mary Poppins</div> <div>Enter opening balance : 2000</div> <div>Output :</div> <div>Account created successfully</div> <div>Enter Choice: 3</div> <div>Enter account number : 1001</div> <div>Enter withdraw amount: 500</div> <div>Output :</div> <div>Your updated balance : 1500</div>

Extra Problems

1. CamelCase
<https://www.hackerrank.com/challenges/camelcase/problem>
2. Caesar Cipher
<https://www.hackerrank.com/challenges/caesar-cipher-1/problem?isFullScreen=true>
3. Alternating Characters
<https://www.hackerrank.com/challenges/alternating-characters/problem?isFullScreen=true>

***Textbook** : B. A. Forouzan and R. F. Gilberg —Cengage Learning , Computer Science: A Structured Programming Approach Using C++ Third Edition.