

Lab instructions

Week 12

Introduction to Programming
ECS 102, 2018-19 Semester II
IISER Bhopal

pointer_strstr.c

The function *strstr*(string1, string2) searches string1 to see whether the string2 is contained in string1, and returns the first occurrence as a character pointer. If there is no occurrence it returns NULL. Write a program to count the number of occurrences of string2 in string1.

- (a) Do not use *strcpy* function
- (b) Do not use *array []* symbol
- (c) Use the function *strstr* only once in each iteration

array_of_pointers.c

Declare an array of three strings of maximum allowable 30 characters each as "India", "United States of America", "Switzerland". What is the total memory allocation in bytes?

Declare an array of three pointers pointing to the first characters of the three strings "India", "United States of America", "Switzerland". What is the total memory allocation in bytes?

- (a) Print the strings both in array and pointer form.
- (b) Increment each string by 5 characters and print the strings both in array and pointer form.
- (c) Increment each string by 6 characters and print the strings both in array and pointer form.

Explain your answers justifying the memory allocation in array and pointer form.

pointer_functions.c

Declare an integer i and pass the address of the integer to a function. Define the function with an argument of $int *$ and increment the pointer by 10 inside the function. The return type of the function should be $void$. Print i after calling the function.

Declare an array of integers and write a function *sort* to sort the array. Do not use the *array []* symbol while using the *sort* function.

Write a function *larger* that takes addresses of two integers and returns the address of the larger of the two integers.

If you return an address of a local variable from a function, see what happens in compile time and run time. Explain your answer.

pointer_to_functions.c

Write a function *table* that takes as arguments a function pointer, start, end, and step. Print *cos* and *sin* between 0 and π with a step $\pi/4$.

Define a user-defined function $y(x) = x^2$ and print it using the same *table* function between 0 and 2 with a step 0.5.

Define a user-defined function $y(x_1, x_2) = x_1^2 + x_2^2$ and print it using a modified *table* function between 0 and 2 with a step 0.5 for both the inputs x_1 and x_2 .