

# Lab instructions

## **Week 11**

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

# use\_pointers.c

Define a pointer variable  $p$  and assign it to the address of an integer  $i$ . Assign  $i = 1$ .

- 1) Print the value and address of  $i$ . Print the pointer  $p$  and  $p$  itself. (You can typecast pointers as “unsigned long int” to avoid warning.)
- 2) Declare another variable  $j$  and assign it to  $*p$ . Assign  $*p = 2$ . Print the values and addresses of both  $i$  and  $j$ . Explain the output.
- 3) Do the same using pointers of short int and long int, and note the differences.
- 4) Verify how many bytes does a pointer variable take by declaring more than one pointer in your code. Does it depend of the data type of the pointer variable? Explain your answer.

# pointer\_inr\_decr.c

You can increment or decrement pointers like variables. Print the pointer  $p$  and  $p$  itself for the following cases.

Increment and decrement using

- 1)  $p++$  and  $p--$
- 2)  $(\text{unsigned long int}) (p + 1)$  and  $(\text{unsigned long int}) (p - 1)$
- 3)  $(\text{unsigned long int}) p + 1$  and  $(\text{unsigned long int}) p - 1$

Explain your answers.

# pointer\_array\_1D.c

Define one 1D array as  $x[3] = \{1, 2, 3\}$ ;

- 1) Print  $x$ . What does it denote?
- 2) Print the elements and their addresses in the array.
- 3) Print the elements and their addresses in the array using a pointer. The pointer should not change by itself.
- 4) Print the elements and their addresses in the array using a pointer by incrementing the pointer itself.

# pointer\_array\_2D.c

Define one 2D array as  $xx[3][3] = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ ;

- 1) Print  $xx$ . What does it denote?
- 2) Print  $xx[m]$  for  $m = 1, 2, 3$ . What do they denote?
- 3) Print the elements and their addresses in the array.
- 4) Print the elements and their addresses in the array using an array of 3 pointers pointing to the 3 rows. The pointers should not change by themselves.
- 5) Print the elements and their addresses in the array using a single pointer by incrementing the pointer itself.