Lab instructions
Week 13

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

pointer_structure.c

Generate the following inventory report. Use a pointer of the type *struct inventory* to read the data from user input. After reading, use the pointer to print the inventory report.

Value = Quantity*Rate and Total in the end is the sum of all the values. Align the data in the columns appropriately.

```
-------------------------------------------------------------------

Code            Quantity             Rate          Value

-------------------------------------------------------------------

F105                 275             237.00

H220                   1             535.50

I109                  52               5.30

M331                   5            1000.00

-------------------------------------------------------------------

                                                Total
```

dma.c

(a) Allocate memory for an integer array dynamically at run-time from the use input of the size of the array. Print the elements of the array and their addresses. Release the memory space at the end and check whether you can access the array after that.

(a) Create a structure student containing name and roll number. Allocate multiple blocks of memory depending on the class size. Populate data for 3 students and print them.

(a) Allocate memory for a string name and print the allocated memory size using *malloc_usable_size* function in *malloc.h*. Reallocate the memory to accommodate a larger length of a string. Allocate different sizes of memory and analyze the output you are getting from the *malloc_usable_size* function.

linked_list.c

Create a linked list of an integer using recursive function. Also print the elements of the linked list and the number of elements using recursive functions.

Insert a number in the linked list at any position.

Delete a number in the linked list at any position.

Create a linked list and insert numbers in the list to keep the numbers always sorted.

stack_queue.c (optional)

Create a doubly linked list with both pointers *prev and *next.

Design a *stack* data structure, where last element goes in come out first.

Design a *queue* data structure, where the first element goes in comes out first.