**National University of Computer and Emerging Sciences**



**Lab Manual # 02**

Object Oriented Programming

|  |  |
| --- | --- |
| Course Instructor | Mr. Farooq Ahmad |
| Lab Instructor (s) | Mr. Dilawar Shabbir  Miss. Hira Chaudhry |
| Section | 3J |
| Semester | Fall 2021 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

# **Objective:**

* Dynamic allocation of arrays using single pointer
* Dynamically growing array (or a vector structure)
* Double pointers and dynamic allocation of 2D arrays (using array of arrays)

|  |  |
| --- | --- |
| **Task 1:** | **Expand Array** |

Write a program that keeps taking integer input from the user until user enters -1 and displays the data in reverse order.

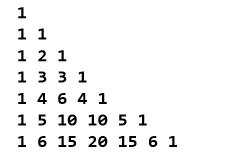
Your program should save the input in a dynamically allocated array. Initially create a dynamic array of five integers. Each time the array gets filled your program should double the size of array (i.e. create a new array of double size, copy previous data in new array, delete previous array) and continue taking the input. After receiving -1 (i.e. end of data input) your program should print the numbers in the reverse order as entered by the user.

**Important Note:** subscript operator [] is not allowed to traverse the array. Use only offset notation. i.e instead of using myArray[i] use \*(myArray+i) to read/write an element. **Do not consume extra space. There shouldn’t be any memory leakage or dangling pointers in your code.**

|  |  |
| --- | --- |
| **Task 2:** | **Dynamic Memory Allocation** |

A ragged array is an array which contains a varying number of elements in each row. The Pascal triangle can be used to compute the coefficients of the terms in the expansion of (a + b) n. Write a function that creates a ragged array representing the Pascal triangle. In a Pascal triangle, each element is the sum of the element directly above it (if any) and the element to the left of the element directly above it (if any).

A Pascal triangle of size 7 is shown below:



You are required to define the following functions:

**Task 2.1:**

**int\*\* createPascalTriangle (int n);**

This function will take an integer (n) as argument and create a Pascal triangle consisting of n rows. It will dynamically allocate the two-dimensional ragged array, fill up its elements, and return a pointer to this filled array (Pascal triangle).

**Task 2.2:**

**void displayPascalTriangle (int\*\* pt, int n);**

This function will take a pointer pt which is pointing to a Pascal triangle consisting of n rows. It will display the Pascal triangle on screen.

**Task 2.3:**

**void deallocatePascalTriangle (int\*\* pt, int n);**

This function will take a pointer pt which is pointing to a Pascal triangle consisting of n rows. This function will deallocate the two-dimensional array containing the Pascal triangle.

**Task 2.4:**

Write a main function which asks the user to specify the value of n. After that the main should call the above functions to create a Pascal triangle of size n, display it on screen, and, finally, de-allocate all the dynamically allocated memory.