LAB-3

OBJECT ORIENTED PROGRAMMING

BSCS-SPRING-2022

Laraib Afzaal

Tasks

QUESTION#1

Write a function that takes two arguments as inputs: an pointer to integer p and size s. You are required to allocate memory for an array with s elements of type int using "new" operator. As your next step, randomly initialize all the elements of the arrays and then update each element value with its square, i.e. replace each element value with its square and print updated array values in main.

Question for your thought why we are passing the pointer by reference into the function?

QUESTION#2

Write a function that takes two arguments as input: an array of pointer ptr. Your goal is to define memory for four individual integer variables using "new" operator and store the addresses of these four variables into the each location of ptr. Next please randomly initialize all the four variables using ptr and find the maximum value and return this value from the function.

QUESTION#3

In this task, we consider ptr as a 2D matrix. Whereas, number of rows would be fixed but make number of columns variables. To do this, instead of using arrays of fixed column size allocated on stack, you will make each element of array to have nColumns (passed as argument to the function) elements and store them on heap using new operator. Next, initialize the values of these arrays randomly using for loop and ptr. Finally display the maximum of each individual array.

QUESTION#4

In the previous task we can consider p as a 2D matrix. However in that case our number of rows (which was 5 as there were 5 arrays stored in p) and number of columns (which were 6 as each array had 6 elements) were fixed. In this question we will extend the previous question and make our number of columns variables. To do this, instead of using arrays of fixed size of 6 elements allocated on stack, we will make each array to have ncol (passed as argument to the function) elements and store them on heap using new operator. Once again store the starting address of each array to array of pointers p. Next, initialize the values of these fives arrays randomly using for loop and p. Finally display the sum of the each individual array. You are not allowed to access the array using original variable name.

QUESTION#5(BONUS)

Write a function that receives three arguments: (i) a 2D pointer p; (ii) number of rows sizeA; (iii) number of columns sizeB and return sum of the edges (boundary rows and columns which are shown as red area in the following figure) of the 2D array. Remember you have to calculate sum of values at edges of the array shown as red area in the given figure only.