Object Oriented Programming Lab

Lab 02 **Marks 10**

Instructions

Work on this lab individually. You can use your books, notes, handouts etc. but you are not allowed to borrow anything from your peer student.

Marking Criteria

Show your work to the instructor before leaving the lab to get some or full credit.

What you must do

Program the following tasks in your C++ compiler and then compile and execute them.

Task 1

Write a program that performs the following tasks.

1. Declare two **integer pointers** named **ptrX** and **ptrY** and initialize them with **NULL**.
2. Create **two variables** with values **2** and **8** on **heap memory segment** and assign their addresses to **ptrX** and **ptrY** respectively.

Now print the following information:

1. The address of **ptrX**, value of **ptrX** and the value of **memory location** where it points to.
2. The address of **ptrY**, value of **ptrY** and the value of **memory location** where it points to.

Free the resources allocated on **heap memory segment**.

Task 2

Write a program that performs the following tasks.

1. Ask the user to enter **size** of a **float array**.
2. Allocate **memory** to an **array** based on the **size** provided by **user**.
3. Initialize the **array content** by **reading** them from **user**.
4. Calculate and display the **average of array elements**.
5. **Free any memory resources** allocated by the program before exit.

Task 3

Implement following **function** named ***getEvenNumbers***

```
int* getEvenNumbers(const int ar[], const int size, int& evenArraySize);
```

The parameters ***ar*** and ***size*** holds **an array** and its **size** respectively.

The function should **return a pointer to newly created array** which contains **only even numbers** exist in array ***ar*** and store its **size** in parameter ***evenArraySize***. It should store **0 (zero)** in ***evenArraySize*** and return **NULL** if ***ar*** contains only **odd** numbers. *The function should **not display** anything.*

In **main** function declare an **array** of **size 10**. Fill the array with arbitrary values and then pass it to ***getEvenNumbers*** function along with its **size** and all the **required parameters**. Display **contents** of the array **returned** by function ***getEvenNumbers*** if any, otherwise display a message *"No Negative Numbers Exist in the Array!"*. Don't forget to free the memory resource allocated by the program, if any.

Task 4

Implement following **function** named ***getPosNeg*** that accept an array ***ar*** along with its size ***n_ar***

```
void getPosNeg(const int ar[], const int n_ar, int* &pos, int& n_pos, int* &neg, int& n_neg);
```

The parameters ***ar*** and ***n_ar*** holds **an array** and its **size** respectively.

The function gets all the **positive** and **negetive** numbers from the array ***ar*** and place them into a **newly created arrays** pointed by parameter ***pos*** and ***neg*** respectively. Store the sizes of ***pos*** and ***neg*** arrays into ***n_pos*** and ***n_neg*** respectively. It should store **0 (zero)** and **NULL** in parameters ***n_pos/n_neg*** and ***pos/neg*** respectively, if ***ar*** has no **positive/negetive** numbers. *The function should **not display** anything.*

In **main** function declare an **array** of **size 10**. Fill the array with arbitrary values and then pass it to ***getPosNeg*** function along with its **size** and all the **required parameters**. After the execution of function display **contents** of the **arrays** pointed by ***pos*** and ***neg***. Display appropriate message(s), if ***pos*** and/or ***neg*** arrays are empty. Don't forget to free the memory resource allocated by the program, if any.

☺ ☺ ☺ **BEST OF LUCK** ☺ ☺ ☺