BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI (RAJ.)
CS F111 Computer Programming

LAB TEST

Instructions

- Make sure that your code is error-free and final script files should run without any warning/ errors.
- The order of execution/ output of the program should be same as mentioned in the provided use cases.
- Submit the complete code in a zipped folder named **<your ID>_Q1.zip**.

Question 1 [30 M]

Assume that you have to create a software for a pizza store. The functioning of the pizza store is organized as per the properties of the pizzas and the store itself. Each pizza is associated with an ID, known as pid (integer), type (integer), price (float), and base (integer). A menu needs to be maintained for pizzas that are available in the store with their details. You are required to use structures to accomplish the desired tasks. A partial code is available in the same folder where a total of six files are present including **"pizza.h"**, **"pizza_prop.c"**, **"pizza_menu.c"**, **"pizza_menuNew.c", "pizza_menu.h"**, and **"main_pizza.c"**.

The file **"pizza.h"** contains the structure definitions that are related to defining a pizza. It also has a few functions which are required to be implemented in **"pizza_prop.c"** as per the specifications given in **"pizza.h"**. Similarly, **"pizza_menu.h"** defines a global array **pizzaMenu** of type **struct pizza** (typdefed as PIZZA) in order to store pizzas. The **MAX_SIZE** has been used to give a maximum size to that array. The variable **count** is used to keep track of number of pizzas present in the **pizzaMenu** array. Recall the concept of addition and deletion of objects from an array by keeping a max size array. You should use that concept over here to implement this. There are some functions declared in **"pizza_menu.h"**, which you should define in **"pizza_menu.c"** and **"pizza_menuNew.c"**. The **main()** function is implemented in **"main_pizza.c"**. This file is complete and you are not required to make any changes to it.

You are required to submit a zipped folder (named **<your ID>_Q1.zip**) with error-free code for the following functionalities:

- Implement all functions in **pizza_prop.c** and **pizza_menu.c** as per the specifications given in the header files along with the function declaration. [6+9]
- The file **pizza_menuNew.c** should follow the same functioning as per **pizza_menu.c** except the **sortPizzaMenuOnPrice()** function. Here, the sorting should done be in a way that the pizza with the highest price should be displayed first. [9]
- Create two different shell script files **myScriptOne.sh** and **myScriptTwo.sh** that compiles and executes the above files (with **pizza_menu.c** and **pizza_menuNew.c** respectively). All inclusive, the file **myScriptOne.sh** should do the intended tasks and print the pizza menu in ascending order, whereas the file **myScriptTwo.sh** should perform the same functioning while printing pizza menu in descending order of their price. [3+3]

Example use-cases are provided below.

Case I (for myScriptOne.sh)	**Case II (for myScriptTwo.sh)**
Provide the number of pizzas to be entered: 4 Enter ID for pizza 1 1001 Enter price for pizza 1 453.26 Enter ID for pizza 2 1002 Enter price for pizza 2 857.93 Enter ID for pizza 3 1003 Enter price for pizza 3 189.91 Enter ID for pizza 4 1004 Enter price for pizza 4 278.45 Pizza menu 1 is: ID: 1001 Type: 2 Price: 453.260010 Base: 2 Pizza menu 2 is: ID: 1002 Type: 1 Price: 857.929993 Base: 1 Pizza menu 3 is: ID: 1003 Type: 1 Price: 189.910004 Base: 3 Pizza menu 4 is: ID: 1004 Type: 3 Price: 278.450012 Base: 2 Pizza Menu after Sorting Pizza menu 1 is: ID: 1003 Type: 1 Price: 189.910004 Base: 3 Pizza menu 2 is: ID: 1004 Type: 3 Price: 278.450012 Base: 2	Provide the number of pizzas to be entered:4 Enter ID for pizza 1 1001 Enter price for pizza 1 453.26 Enter ID for pizza 2 1002 Enter price for pizza 2 857.93 Enter ID for pizza 3 1003 Enter price for pizza 3 189.91 Enter ID for pizza 4 1004 Enter price for pizza 4 278.45 Pizza menu 1 is: ID: 1001 Type: 2 Price: 453.260010 Base: 2 Pizza menu 2 is: ID: 1002 Type: 1 Price: 857.929993 Base: 1 Pizza menu 3 is: ID: 1003 Type: 1 Price: 189.910004 Base: 3 Pizza menu 4 is: ID: 1004 Type: 3 Price: 278.450012 Base: 2 Pizza Menu after Sorting Pizza menu 1 is: ID: 1002 Type: 1 Price: 857.929993 Base: 1 Pizza menu 2 is: ID: 1001 Type: 2 Price: 453.260010 Base: 2

Pizza menu 3 is: ID: 1001 Type: 2 Price: 453.260010 Base: 2 Pizza menu 4 is: ID: 1002 Type: 1 Price: 857.929993 Base: 1	Pizza menu 3 is: ID: 1004 Type: 3 Price: 278.450012 Base: 2 Pizza menu 4 is: ID: 1003 Type: 1 Price: 189.910004 Base: 3