**44-542 Object Oriented Programming**

**Lab03: Control Structures Lab Activity**

**Objective:** Covers the usage of **Conditions, Operators**, **Selections, and Repetitions.**

**NOTE:**

* Do not hard code any values.
* Check the sample output to know how the results need to be printed.
* Read every instruction carefully and follow them strictly.
* Do not change the name of the attributes, and methods given below.

1. Create a New Project and name it as **Lastname\_Lab03ControlStructures** where **Lastname** is your last name.
2. Create a new package and name it as **pizzaorder.** All classes in this lab should be in a package named **pizzaorder.**
3. Create a new java class, and name it as **PizzaOrder**. This class provides methods for calculating final price of the pizza.
4. Attributes are given in this table. You must use the names given here, with the same capitalization as shown here. All attributes are private.

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Attribute Description** |
| **name** | **String** | Name of the Customer |
| **address** | **String** | Address of the Customer |
| **contactNumber** | **String** | Contact number of the customer |
| **size** | **char** | Size of the pizza. It is either ‘S’ or ‘M’ or ‘L’. |
| **BASIC\_CRUST\_COST** | **double** | It’s a constant (final) with value 4.0, defines cost of the pizza crust |
| **BASIC\_CHEESE\_COST** | **double** | It’s a constant (final) with value 2.0, defines cost of the cheese added |
| **BASIC\_SAUCE\_COST** | **double** | It’s a constant (final) with value 0.50, defines cost of the sauces added |
| **DELIVERY\_CHARGE** | **double** | It’s a constant (final) with value 2.75, defines delivery cost |
| **toppings** | **String** | Toppings on the pizza separated by a comma. |

1. Constructor: There will be only one constructor. It has the following signature:

**public PizzaOrder(String name, String address, String contactNumber, char size, String toppings).**

1. Methods: Methods are described below. All methods are public.
   1. Write getters for the given private instance variables.
   2. **getDELIVERY\_CHARGE():** Returns the delivery charge of the pizza.
   3. **getNumberOfToppings(**): This method identifies the number of toppings present in the **String** **toppings**. For example, if **toppings** contains “green peppers, jalapenos, grilled chicken”, then this method must return 3.

(**HINT**: Use comma along with **String** functions to calculate the number of toppings).

* 1. **getFinalCrustCost():** This method returns the cost of the pizza crust based on the **size** of the pizza crust selected by the customer. Use switch statement and calculate, if **size** is small -‘S’, then final crust cost will be **BASIC\_CRUST\_COST**. If **size** of the crust is medium -‘M’, then final crust cost is 30% more than **BASIC\_CRUST\_COST**. If size of the crust is large (default) crust cost is 60% more than **BASIC\_CRUST\_COST**.
  2. **getFinalCheeseCost():** This method returns the cost of the cheese added on the pizza based on the **size** of the pizza. If **size** is small – ‘S’, then returned value will be **BASIC\_CHEESE\_COST**. If **size** of the crust is medium – ‘M’, then returned value will be 40% more than **BASIC\_CHEESE\_COST**. If **size** of the crust is large (default), then returned value will be 80% more than **BASIC\_CHEESE\_COST**.
  3. **getToppingCost():**This method returns the total cost of the toppings in a pizza based on the number of toppings and **size**. Cost of a topping varies according to the size of the pizza ordered.

|  |  |
| --- | --- |
| **Size** | **Cost of one topping($)** |
| **Small(S)** | **1.5** |
| **Medium(M)** | **2.0** |
| **Large(L)** | **2.5** |

If the number of toppings is 3 and **size** of the pizza is M, then total toppings cost is 6$.

* 1. **getFinalPrice(char deliveryOption):**This method returns the final price of the pizza, along with the delivery charges( if selected by the user). **deliveryOption** either contains a ‘D’ or ‘P’. If ‘D’ option is selected final price includes crust cost, cheese cost, toppings cost, sauce cost and delivery charge. If delivery cost is not selected then final price will contain only crust cost, cheese cost, toppings cost, sauce cost.
  2. **toString() method:** This method returns a string representation of the pizza object.

*Example: For the pizza described above, the* **toString** *method would return the following string:*

|  |
| --- |
| *Order Name: order1*  *Address: 1115 North College Drive, Apt #222, Maryville*  *Contact Number: 6602481234*  *pizza size: S*  *Number of toppings: 2*  *Crust cost: $4.0*  *Sauce cost: $0.5*  *Cheese cost: $2.0*  *Toppings cost: $3.0* |

1. Include Javadoc comments, using the @author, @param, and @return annotations when appropriate.
2. Generate documentation for your project by clicking on **Run** from the NetBeans menu bar and then selecting **Generate Javadoc**. The documentation will be placed in a **javadoc** subfolder of the **dist** subfolder inside your project folder. To view the documentation, open the **index.html** file that is created.
3. Create a class named **PizzaOrderDriver.** This class has a method **main** and uses one **Scanner** object to get input from user console.
4. Here is a pseudocode version of this class. You can use different variable names than those shown here. You do not have to follow these steps exactly, but you must produce the same output (**Look at sample Output**), *including labels*.

|  |
| --- |
| *do{*  *Declare and initialize a scanner object sc.*  *Enter the delivery option either pick up or delivery. Scan the delivery option entered by the user. If delivery option selected is ‘D’ scan the address entered by the user.*  *Scan the contact details of the customer entered.*  *Scan the size of the pizza requested by the customer(S/M/L)*  *Scan the topping requested by the customer*  *Invoke the pizzaOrder constructor based on the scanned details.*  *Print the details using toString() method.*  *If{*  *Test the delivery charge method if delivery option is selected.}*  *Test the final price of the pizza*  *Prompt the customer whether an additional order is required.*  *}*  *While{*  *If he chooses Yes order form must repeat else program must terminate with a Thank you message*  *}* |

**Sample Output may look like this (Input in Red):**

|  |
| --- |
| Welcome to Northwest Pizza!  Please enter a name for order: CSIS  Please choose Delivery(D) or Pick up(P): D  Please enter your address: 800 University Drive  Please enter your contact number: 6602481234  Please choose the size of pizza(S/M/L): M  Please enter the toppings you want to add separated by a comma: green peppers, red peppers, onions  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Please review your order below:  Order Name: CSIS  Address: 800 University Drive  Contact Number: 6602481234  Pizza size: M  Number of toppings: 3  Crust cost: $5.2  Sauce cost: $0.5  Cheese cost: $2.8  Toppings cost: $6.0  Delivery charges: $2.75  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Total bill amount to be paid: $17.25  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Do you wish to place another order?(Y/N):  Y  Please enter a name for order: CSIS-2  Please choose Delivery(D) or Pick up(P): P  Please enter your contact number: 6602481234  Please choose the size of pizza(S/M/L): L  Please enter the toppings you want to add separated by a comma: chicken, tomato, mushrooms  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Please review your order below:  Order Name: CSIS-2  Address: Not given  Contact Number: 6602481234  Pizza size: L  Number of toppings: 3  Crust cost: $6.4  Sauce cost: $0.5  Cheese cost: $3.6  Toppings cost: $7.5  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Total bill amount to be paid: $18.0  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Do you wish to place another order?(Y/N):  N  Thank you for your order! |

**Submit you solution by following the steps below:**

* Save your files in NetBeans.
* Zip your entire Project. (It should be called **Lastname\_Lab03ControlStructures**.zip where Last name is your last name.)
* Submit the Zip file to the Lab03ControlStructures dropbox.
* Download the Zip file you have submitted.
* Look in the Zip file and verify the class files, javadocs in the Zip folder are updated. If not, resave your project in NetBeans and resubmit.