Project Two Pseudocode

Main_Function {

Lets create two arrays to store regular customer and Preferred customers.

Customer[] array_regular = new Customer[10]// we begin with array of size 10 but we will resize the array as needed.
Customer[] array_preferred = null // we haven't initialized the array as we may not need it

Read regular customer file and fill the array called array_regular accordingly. Multiple resizing might be necessary.

Read Preferred customer file and fill the array called array_preferred accordingly. Multiple resizing might be necessary. Gold and Platinum customers will be inserted into respective Gold and Platinum customer objects.

Preferred Customer file might not exists, work accordingly

Start working with Orders file

For every line in orders file {
Check the line has the correct number of items, if not print error message with line number and go to the next line

Check if each item is of the correct type, if not print error message with line number and go to the next line

Process the current order, changing the customer details as per order.

First find the customer in regular customer array or preferred customer array using **check id** method.

Calculate the price of the order using **calculate_price** routine

Calculate the new amount spend for the customer.

If the customer has entered a new status, then according change the customer object. Might need to change regular customer to preferred or Gold to Platinum. Apply any new discount for the customer and accordingly change the amount spent as well. Refer the cases to code and test section.

```
      This might require remove_customer method and put_customer methods.

      }// end of processing order file
      Copy regular customer array_regular[] into customer.dat file.
      Copy preferred customer array array_preferred[] info into preferred.dat file.

}end of Main_Funtion
```

Other Functions in Main.java

check_id

This method checks if the id matches any customer inside the array if so return the customer object if not return null

```
Customer check_id(String id, Customer[] array) {
          For every index in the array {
             Check if the id matches the customer at that index
                   If so return array element;

          }
          return null; // customer not found in the list

      }
```

Remove_Customer

This method will remove the customer from the array and resize the array

```
Customer[] remove_customer(Customer cus, Customer[] array) {

      Customer[] new_array = new Customer[array.length-1];
           int i, k=0; // i will keep track of index of array and k will be track of
index of new_array
             for( i=0; i < array.length; i++) {
                   if(!(cus.equals(array[i]))){
                         new_array[k] = array[i]
                         k++
                   }
             }


        return new_array;



      }
```

put_Customer

This method will put a customer into the array after resizing the array. returns the new array reference

```
Customer[] put_customer(Customer cus, Customer[] array) {

      if (array==null) {
            Customer[] new_array = new Customer[1];
            new_array[0] = cus;
        return new_array;

      }
      Customer[] new_array = new Customer[array.length+1];
    arraycopy(array, 0, new_array, 0, array.length);
    new_array[array.length] = cus;
    return new_array;


}
```

calculate_price

This method will calculate the price of a drink based on its type, size, graphic amount and quantity

```
float calculate_price(String Size, String Type, float inch_p , int quantity) {


      final float soda_price = (float)0.2, tea_price =  (float)0.12 ,
punch_price = (float)0.15; // price per ounce for drink type

      //Lets initialize Cylinder sizes
      final float small_diameter =  (float)4, small_height = (float) 4.5;
      final float medium_diameter =  (float)4.5, medium_height = (float) 5.75;
      final float large_diameter =  (float)5.5, large_height = (float) 7;
      final int small = 12, medium = 20, large = 32;  // liquid quantity for
each drink size
```

The price of a single drink is the sum of price of the drink and price for the graphic on the cup, which will depend on the area of the cylinder

```
float drink_price, cup_price, size=0, price_per_oz=0, cyl_surface_area =0;
```

To calculate the drink_price, if need to know the size and type of the drink. For each type the price per oz is different.

```
      drink_price = size * price_per_oz;
```

To calculate the price of the cylinder we need to calculate its surface area
Formula for cylinder surface area = 2*pi*radius *height or diameter*height*pi
We need to insert correct diameter and height based on size of the drink

```
cyl_surface_area =  (float)( diameter of the cup * height of the cup * Math.PI);
```

Now that we have calculated the surface area lets calculate the price of the cup

```
cup_price = cyl_surface_area * Graphic_cost_per_inch;
```

Now for the total price we will calculate the total per drink and multiple it with the quantity

```
float total_drink_price = cup_price + drink_price;

Finally, the price is price per drink multiplied by the quantity
return total_drink_price * quantity;

}
```

Cases to check and code

1. A regular customer places an order. The amount spent is less than 50 after processing the order. Customer's amount spent is updated. The status of the customer does not change.

2. A regular customer places an order. The amount spend is more than 50 but less than 100. 5% discount is applied to the order and the amount spend is changed accordingly. The customer status changes to Gold with 5% discount. The customer is removed from regular customer array and inserted to preferred customer array.

3. A regular customer places an order. The amount spend is more than 100 but less than 150. 10% discount is applied to the order and the amount spend is changed accordingly. The customer status changes to Gold with 10% discount. The customer is removed from regular customer array and inserted to preferred customer array.

4. A regular customer places an order. The amount spend is more than or equal to 150. 15% discount is applied to the order and the amount spend is changed accordingly.

    a. If the amount spend after the discount is more than or equal to 200 The customer status changes to Platinum with computed bonus bucks. The customer is removed from regular customer array and inserted to preferred customer array.

b. If the amount spent is less than 200 , the customer status changes to Gold with 15% discount. The customer is removed from regular customer array and inserted to preferred customer array.

5. A Gold customer with 5% discount places an order. After applying 5% discount, the amount spend is still less than 100. The customer status does not change.

6. A Gold customer with 5% discount places an order. The amount spend is more than 100 but less than 150. 10% discount is applied to the order and the amount spend is changed accordingly. The customer status changes to Gold with 10% discount.

7. A Gold customer with 5% discount places an order. The amount spend is more than or equal to 150. 15% discount is applied to the order and the amount spend is changed accordingly.

    a. If the amount spend after the discount is more than or equal to 200 The customer status changes to Platinum with computed bonus bucks. Customer type changes in preferred array. Do not change Customer index in the array.
    b. If the amount spent is less than 200, the customer status changes to Gold with 15% discount.

8. A Gold customer with 5% discount places an order. The amount spend is more than 200. 15% discount is applied to the order and the amount spend is changed accordingly. Bonus bucks are calculated for every 5 dollars spend after 200. The customer status changes to Platinum with computed bonus bucks.

9. A Gold customer with 10% discount places an order. After applying 10% discount, the amount spend is still less than 150. The customer status does not change.

10. A Gold customer with 10% discount places an order. The amount spend is more than or equal to 150. 15% discount is applied to the order and the amount spend is changed accordingly.

    a. If the amount spend after the discount is more than or equal to 200 The customer status changes to Platinum with computed bonus bucks. Customer type changes in preferred array. Do not change Customer index in the array.
    b. If the amount spent is less than 200, the customer status changes to Gold with 15% discount.

11. A Gold customer with 15% discount places an order. After applying 15% discount, the amount spend is still less than 200. The customer status does not change.

12. A Gold customer with 15% discount places an order. 15% discount is applied to the order and the amount spend is changed accordingly. The amount spent is more than or equal to 200. Bonus

bucks are calculated for every 5 dollars spend after 200. The customer status changes to Platinum with computed bonus bucks. Customer type changes in preferred array. Do not change Customer index in the array.

13. A platinum customer places an order. The current bonus bucks are subtracted from the order total and the amount spend is calculated accordingly. For every 5 dollars spent a bonus buck is added. The customer status does not change.