**(SECJ1013) PROGRAMMING TECHNIQUE 1**

**SEM 1, SESSION 2023/2024**

**LAB EXERCISE 3 (4%)**

**INSTRUCTIONS TO THE STUDENTS**

* This exercise must be done **individually**.
* Any form of plagiarism is **NOT ALLOWED**. Students who copied other students' assignments will get **ZERO** marks (both parties, students who copied, and students who shared their work).

**SUBMISSION PROCEDURE**

* Please submit this exercise no later than **December 21, 2023, Thursday (1 PM MYT)**.
* Only hardcopy is accepted for this submission with handwriting (at my office – n28a, level 2, room 02-31-01).

Fill your information here:

|  |  |
| --- | --- |
| Name |  |
| Matric Number |  |
| Year / Course |  |
| Section |  |

# SET 1

The following program code has errors. Locate the errors.

|  |  |
| --- | --- |
| Line | C++ Codes |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72 | #include <iostream>  using namespace std;  // function prototypes  void display\_Question()  void yes\_No()  int get\_Status()  // start main function  int main() {  char red\_zone, close\_contact, fever;  // two possible character values only:  // 'y' -> yes, 'n' -> no    int status;  // 0 -> GREEN, 1 -> YELLOW, 2 -> ORANGE, 3 -> RED  for (int i = 1; i < 3; i++) {  display\_question(i);  if (i = 1)  yes\_No(red\_zone);  // set red\_zone either 'y' or 'n'  else if (i = 2)  yes\_No(close\_contact);  // set close\_contact either 'y' or 'n'  else  yes\_No(fever);  // set fever either 'y' or 'n'  }  // get risk status based on red\_zone, close\_contact, fever parameters  status = get\_Status(red\_zone, close\_contact, fever);  cout << "Your Covid-19 risk status is ";  while (status) {  if status = 0: cout << "GREEN"; break;  if status = 1: cout << "YELLOW"; break;  if status = 2: cout << "ORANGE";  if status = 3: cout << "RED";  }  cout << "\n";  return 0;  }  // start new user-defined functions  void display\_Question(char q) {  switch (q)  case 1:  cout << "Living in red zone?\n";  break;  case 2:  cout << "Have a close contact with Covid-19 patient?\n";  break;  case 3:  cout << "Body temperature >= 38 degrees Celcius?\n";  } return q;  }  void yes\_No(char ans) {  do {  cout << "Please enter your answer (y / n): ";  cin << ans;  } while (ans = 'n' & ans = 'y');  cout << '\n';  }  int get\_Status(char rz, char cc, char f) {  int s = 0;  if (rz = 'y') s++;  if (cc = 'y') s++;  if (f = 'y') s++;  return s;  } |

Fill in the following table by stating the line number and write the correct statement with the reason(s).

| **Line Number** | **Correct Statement with the reason(s)** |
| --- | --- |
|  |  |

**SET 2**

Complete the code segments in the program below.

1. Task 1:

Write a function named **setType** with the parameter of car type variable,

which prompts the user to enter a car type either "**sedan**" or "**mpv**"

and continues to do so in a loop until the entered type is either "**sedan**" or "**mpv**".

1. Task 2:

Write a function named **setPackage** with the parameter of car wash service package,

which displays a menu with three options for car wash service packages: **Basic (1)**, **Deluxe (2)**, an

d **Premium (3)**.

It prompts the user to choose a package by entering the corresponding number (**1**, **2**, or **3**).

The loop continues until a valid package number (between **1** and **3** inclusive) is entered by the user.

1. Task 3:

Write a function named **wash** with the parameter of car type variable,

which calculates the wash service charge based on the type of car.

If the car type is "**mpv**", the charge is calculated as **1.2 times** the constant **WASH**;

otherwise, the charge is equal to the constant **WASH**.

The calculated charge is then displayed, and the function returns the computed charge.

1. Task 4:

Write a function named **vacuum** with the parameter of car type variable,

which calculates the vacuum service charge based on the type of car.

If the car type is "**mpv**", the charge is calculated as **1.05 times** the constant **VACUUM**;

otherwise, the charge is equal to the constant **VACUUM**.

The calculated charge is then displayed, and the function returns the computed charge.

1. Task 5:

Write a function named **polish** with the parameter of car type variable,

which calculates the polish service charge based on the type of car.

If the car type is "**mpv**", the charge is calculated as **1.2 times** the constant **POLISH**;

otherwise, the charge is equal to the constant **POLISH**.

The calculated charge is then displayed, and the function returns the computed charge.

1. Task 6:

List all function prototypes.

1. Task 7:
2. Call the functions from Task 1 until Task 5 in the **main** function.
3. **totalCharge** is the variable to hold the total service charge based on different wash service package with different car type ("**sedan**" or "**mpv**").

(Note for wash service packages: Task 3 for **Basic**, Task 4 for **Deluxe**, and Task 5 for **Premium**)

1. Print out the final total service charge.
2. Task 8:

You must ensure your program fulfil the following criteria:

* The program is able to run.
* The program uses an appropriate structure for the program (e.g. all required header files are included, the program is properly written, proper indentation, etc.).

**Sample Execution Output**

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |

Note: show user’s input.

|  |
| --- |
| #include <iostream>  #include <string>  using namespace std;  // constants with the associated values  #define WASH 10.0 // the price of WASH service charge  #define VACUUM 7.0 // the price of VACUUM service charge  #define POLISH 15.0 // the price of POLISH service charge  // Task 6: List all function prototypes.  // Task 7: (i) Call the functions from Task 1 until Task 5.  int main() {  string carType; // car type variable  int wsPkg; // car wash service package  float totalCharge = 0; // total service charge based on different wash service package with different car type    // call setType function with the parameter carType      // call setPackage function with the parameter wsPkg        // Task 7: (ii) totalCharge is the variable to hold the total service charge based on different wash service package with different car type (sedan or mpv).  switch (wsPkg) {                      }  cout << endl;  cout << "Total service charge is " << totalCharge;  return 0;  }  // Task 1: Function to set car type  void setType(string &type) {  cout << endl;  }  // Task 2: Function to set wash service package  void setPackage(int &pkg) {  cout << endl;  }  // Task 3: Function to determine exterior wash service charge based on car type  float wash(string type) {  }  // Task 4: Function to determine interior vacuum service charge based on car type  float vacuum(string type) {  }  // Task 5: Function to determine exterior polish service charge based on car type  float polish(string type) {  } |