CS127-5L: Computer Programming 2 Laboratory
Machine Problem #2: Introduction to Classes

Name:	**Carreon, Ma. Addine Anne T.**	**Score:**
Section:	**A35**	**Date: 03-02-2023**

Instructions:

1. Save your file as Surname_Firstname_MP2. Ex. Isip_MP2.cpp

2. You will submit the following and send it to BB.

   a. PDF file of Machine Problem 1 provided with the screenshot of your answers (Sample Run)
   b. C++ script with .cpp extension.

3. Your program must have comments for each section.
   Header Comments:
   Write a description of the program.
   Written by: Cheryl Mari M. Isip
   Date: March 02,2023
   Time: 7:30am
   Program: BSCPE
   Course: CS127-5L
   Section: B20
   School: Mapua University

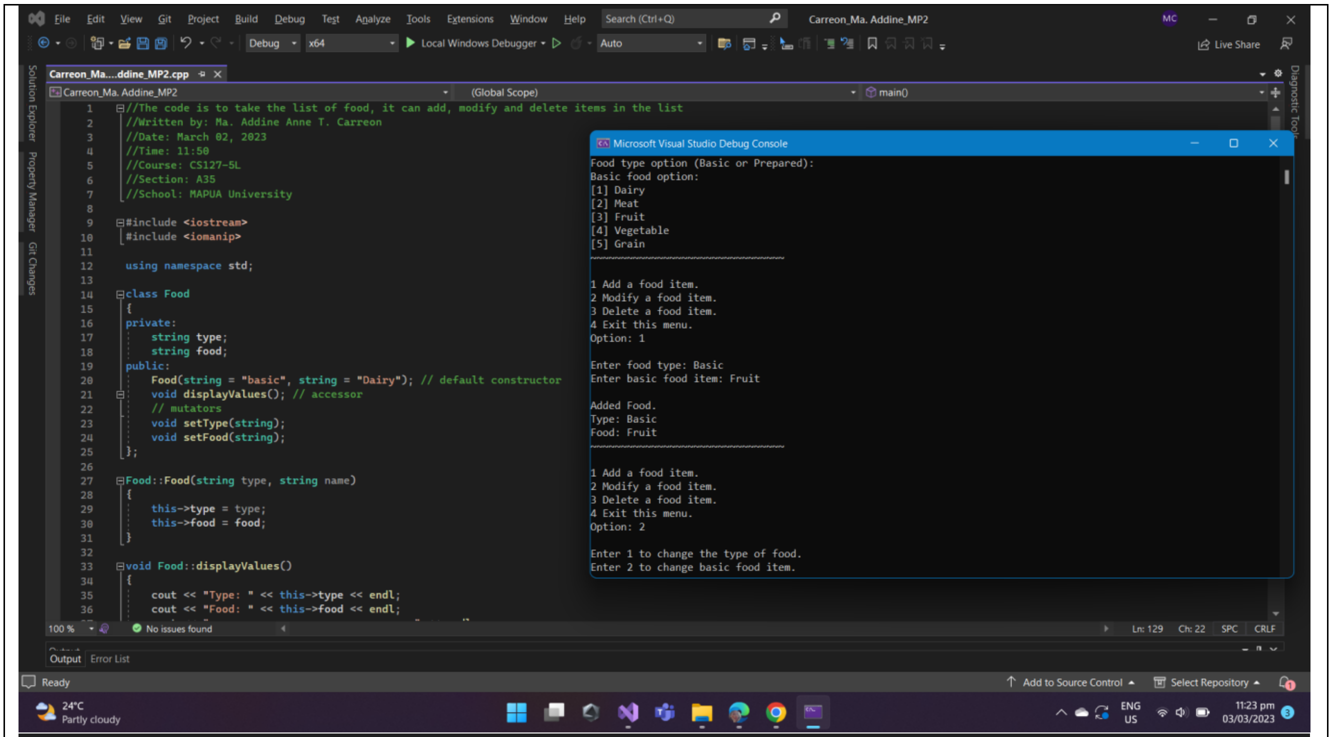
EXERCISE

a. Construct a class definition to represent types of food. A type of food is classified as basic or prepared. Basic foods are further classified as Dairy, Meat, Fruit, Vegetable or Grain. The services the class provides should be the capability to enter data for a new food, the capability to change data for a new food, and the capability to display existing data for a new food.
b. Include the class definition created in a working C++ program that asks the user to enter data for four food items and then displays the entered data.
c. Modify the program written to include a menu that offers the user the following choices:
   1. Add a food item
   2. Modify a food item
   3. Delete a food item
   4. Exit this menu.

Note that In response to the user's choice, the program should initiate an action to implement the choice.

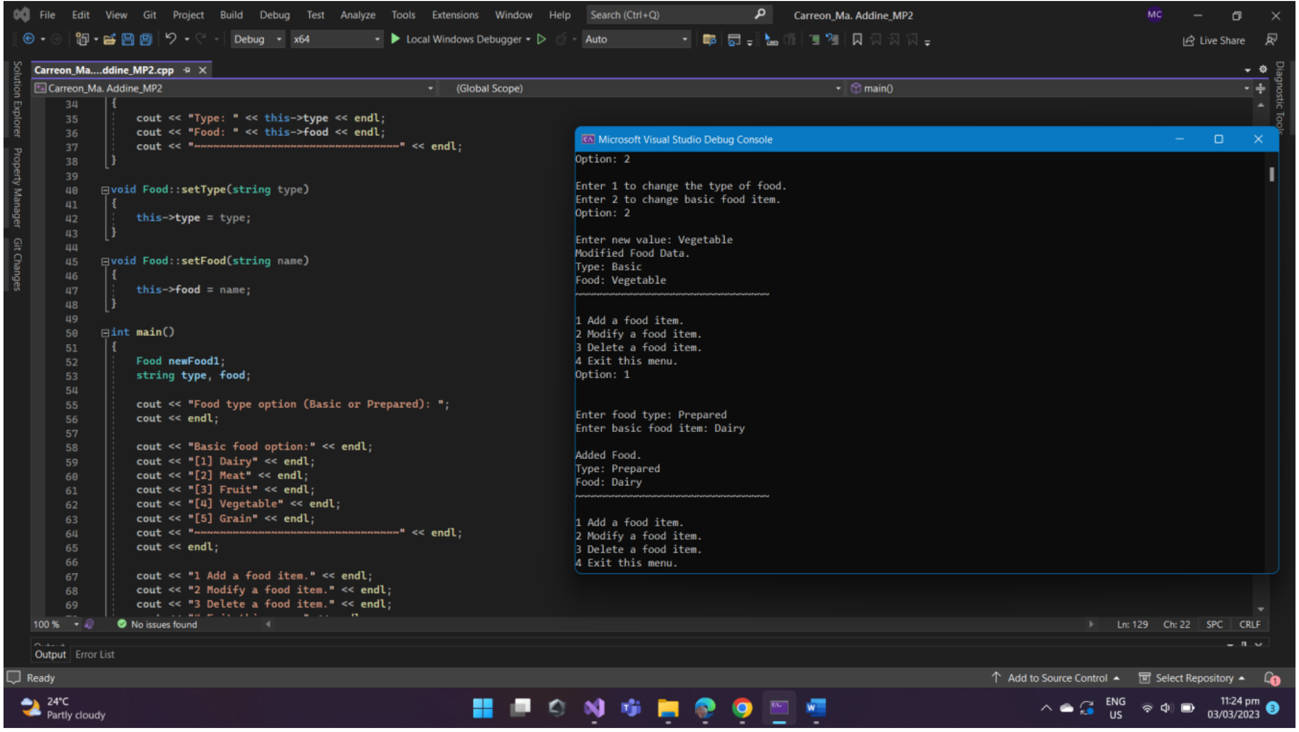
CS127-5L: Computer Programming 2 Laboratory
Machine Problem #2: Introduction to Classes

Take a screenshot and paste your output:



```cpp
//The code is to take the list of food, it can add, modify and delete items in the list
//Written by: Ma. Addine Anne T. Carreon
//Date: March 02, 2023
//Time: 11:50
//Course: CS127-5L
//Section: A35
//School: MAPUA University

#include <iostream>
#include <iomanip>

using namespace std;

class Food
{
private:
    string type;
    string food;
public:
    Food(string = "basic", string = "Dairy"); // default constructor
    void displayValues(); // accessor
    // mutators
    void setType(string);
    void setFood(string);
};

Food::Food(string type, string name)
{
    this->type = type;
    this->food = food;
}

void Food::displayValues()
{
    cout << "Type: " << this->type << endl;
    cout << "Food: " << this->food << endl;
```

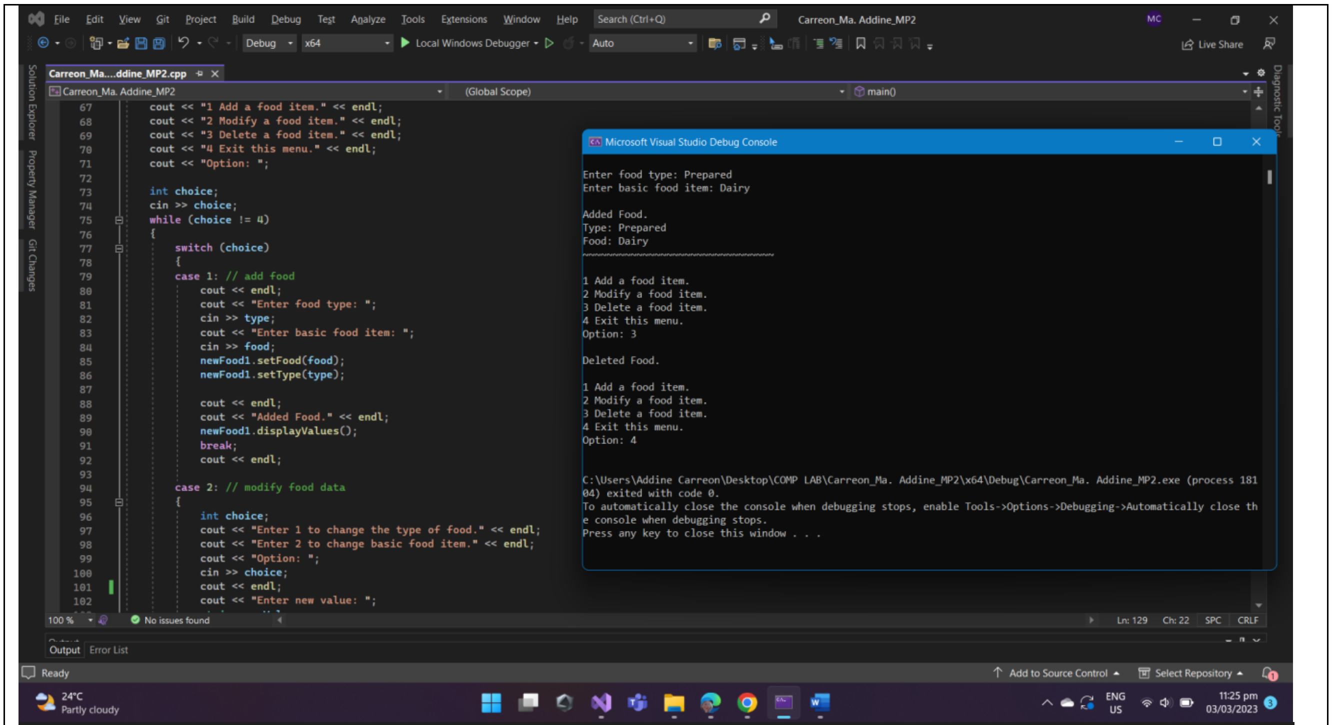
```
Food type option (Basic or Prepared):
Basic food option:
[1] Dairy
[2] Meat
[3] Fruit
[4] Vegetable
[5] Grain
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 1

Enter food type: Basic
Enter basic food item: Fruit

Added Food.
Type: Basic
Food: Fruit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 2

Enter 1 to change the type of food.
Enter 2 to change basic food item.
```



```cpp
{
    cout << "Type: " << this->type << endl;
    cout << "Food: " << this->food << endl;
    cout << "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~" << endl;
}

void Food::setType(string type)
{
    this->type = type;
}

void Food::setFood(string name)
{
    this->food = name;
}

int main()
{
    Food newFood1;
    string type, food;

    cout << "Food type option (Basic or Prepared): ";
    cout << endl;

    cout << "Basic food option:" << endl;
    cout << "[1] Dairy" << endl;
    cout << "[2] Meat" << endl;
    cout << "[3] Fruit" << endl;
    cout << "[4] Vegetable" << endl;
    cout << "[5] Grain" << endl;
    cout << "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~" << endl;
    cout << endl;

    cout << "1 Add a food item." << endl;
    cout << "2 Modify a food item." << endl;
    cout << "3 Delete a food item." << endl;
```

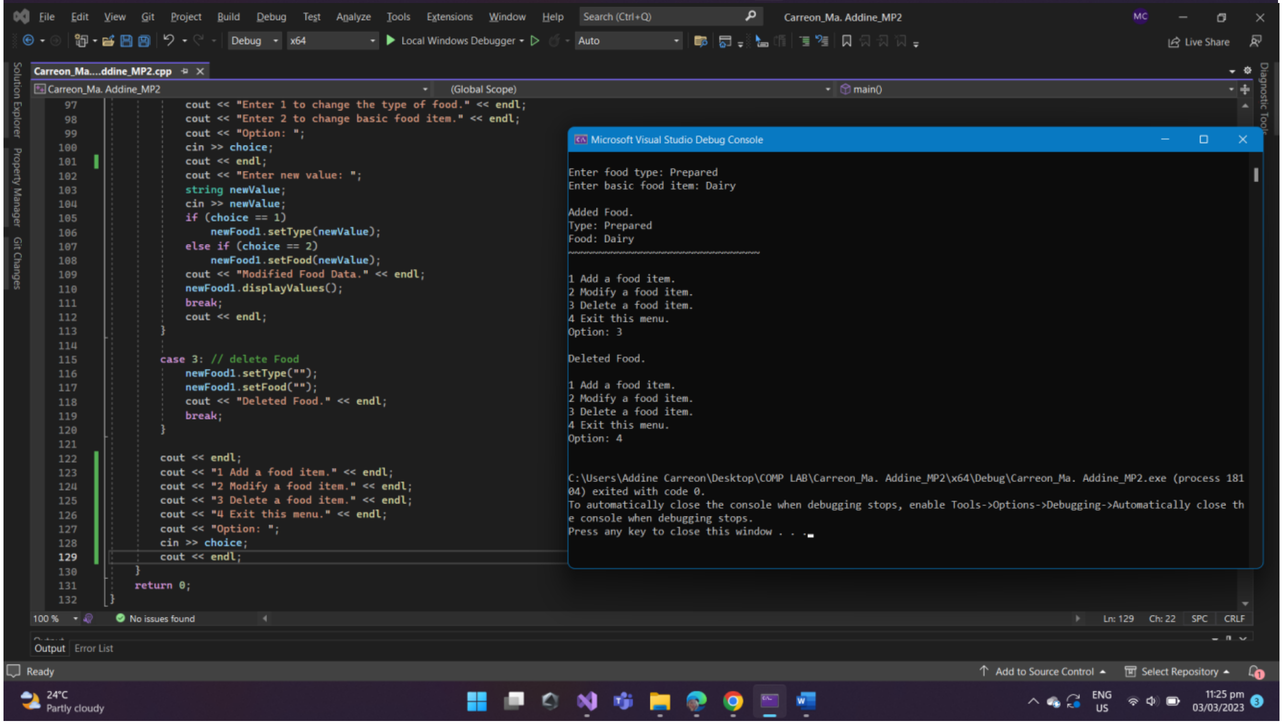
```
Option: 2

Enter 1 to change the type of food.
Enter 2 to change basic food item.
Option: 2

Enter new value: Vegetable
Modified Food Data.
Type: Basic
Food: Vegetable
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 1


Enter food type: Prepared
Enter basic food item: Dairy

Added Food.
Type: Prepared
Food: Dairy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
```

CS127-5L: Computer Programming 2 Laboratory
Machine Problem #2: Introduction to Classes



```cpp
    cout << "1 Add a food item." << endl;
    cout << "2 Modify a food item." << endl;
    cout << "3 Delete a food item." << endl;
    cout << "4 Exit this menu." << endl;
    cout << "Option: ";

    int choice;
    cin >> choice;
    while (choice != 4)
    {
        switch (choice)
        {
        case 1: // add food
            cout << endl;
            cout << "Enter food type: ";
            cin >> type;
            cout << "Enter basic food item: ";
            cin >> food;
            newFood1.setFood(food);
            newFood1.setType(type);

            cout << endl;
            cout << "Added Food." << endl;
            newFood1.displayValues();
            break;
            cout << endl;

        case 2: // modify food data
        {
            int choice;
            cout << "Enter 1 to change the type of food." << endl;
            cout << "Enter 2 to change basic food item." << endl;
            cout << "Option: ";
            cin >> choice;
            cout << endl;
            cout << "Enter new value: ";
```

```
Enter food type: Prepared
Enter basic food item: Dairy

Added Food.
Type: Prepared
Food: Dairy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 3

Deleted Food.

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 4


C:\Users\Addine Carreon\Desktop\COMP LAB\Carreon_Ma. Addine_MP2\x64\Debug\Carreon_Ma. Addine_MP2.exe (process 181
04) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close th
e console when debugging stops.
Press any key to close this window . . .
```



```cpp
            cout << "Enter 1 to change the type of food." << endl;
            cout << "Enter 2 to change basic food item." << endl;
            cout << "Option: ";
            cin >> choice;
            cout << endl;
            cout << "Enter new value: ";
            string newValue;
            cin >> newValue;
            if (choice == 1)
                newFood1.setType(newValue);
            else if (choice == 2)
                newFood1.setFood(newValue);
            cout << "Modified Food Data." << endl;
            newFood1.displayValues();
            break;
            cout << endl;
        }

        case 3: // delete Food
            newFood1.setType("");
            newFood1.setFood("");
            cout << "Deleted Food." << endl;
            break;
        }

        cout << endl;
        cout << "1 Add a food item." << endl;
        cout << "2 Modify a food item." << endl;
        cout << "3 Delete a food item." << endl;
        cout << "4 Exit this menu." << endl;
        cout << "Option: ";
        cin >> choice;
        cout << endl;
    }
    return 0;
}
```

```
Enter food type: Prepared
Enter basic food item: Dairy

Added Food.
Type: Prepared
Food: Dairy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 3

Deleted Food.

1 Add a food item.
2 Modify a food item.
3 Delete a food item.
4 Exit this menu.
Option: 4


C:\Users\Addine Carreon\Desktop\COMP LAB\Carreon_Ma. Addine_MP2\x64\Debug\Carreon_Ma. Addine_MP2.exe (process 181
04) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close th
e console when debugging stops.
Press any key to close this window . . .
```