

Object Oriented Programming 11206 First Exam, Spring 2016/2017

Instructors:

Dr. Raad Qassas

Dr. Nailah Al-Madi

Dr. Anas Abu Talib

Student Name

Student Number

Section Number /Doctor

Question	Points	Score
1	40	
2	30	
3	30	
Total	100	

12/3/2017

Question 1 (40 points):

A remote control unit is used to control a toy car. The remote control unit can open any one of the car's windows. In addition, the remote control unit can move the car forward or backward. Finally, the car wheels can be set to move to the left or to the right with a maximum degree of 60 in either direction.

Write a C++ class to represent the remote control Unit. Call **RemoteUnit**. Within the class, define the following:

1. Member variables to reflect the following:
 - a. Four variables of type bool to reflect (يعكس) the status (حالة) of the four windows of the car. **LeftFront, rightFront, leftBack, and rightBack**. A true value means the window is open.
 - b. An integer member named **status** that reflects the moving status of the car. +1 means moving forward, -1 means backward, and 0 means it is not moving.
 - c. A double value named **angle** that represents the angle of the wheel. The default value 0 means that the wheels are pointing forward. The value -45 means that the wheels have been directed an angle of 45 degrees to the left. The value 60 means that the wheels have been directed an angle of 60 degrees to the right.
2. **A default constructor** that initializes the **RemoteUnit** of data members such that the angle is 0, the status is 0, and the windows are all closed.
3. A member function named **void OpenCloseFL (bool w)**, this function will open the front left window of the car if w is true, and close it if w is false.
4. A member function **isFLOpen()** which returns true if the front left window is open, and returns false otherwise.
5. A member function **MoveForward()** which changes the car status to be moving forward.
6. A member function **TurnWheels(bool left, double degree)** which turns the wheels left or right with the given degree. It will turn left if the first parameter was true. Remember, it is not supposed to turn more than 60 degrees in any direction.

```

#include <iostream>
using namespace std;
class RemoteUnit {
public:
    RemoteUnit() {
        leftFront = rightFront = leftBack = rightBack = false;
        Status = 0;
        Angle = 0;
    }
    void OpenCloseFL (bool w) {
        leftFront = w;
    }
    bool isFLOpen() {
        return leftFront;
    }
    void MoveForward() {
        Status = 1;
    }

    void TurnWheels( bool left, double degree) {
        degree = (degree>=0)?degree:0;
        if (left)
            degree *= -1;
        Angle += degree;
        Angle = (Angle>60)?60:Angle;
        Angle = (Angle<-60)?-60:Angle;
    }
    void MoveBackward() { Status = -1; }
    void StopMoving() { Status = 0; }

private:
    int Status;
    double Angle;
    bool leftFront, rightFront, leftBack, rightBack;
};

```

Question 2 (30 points):

Assume that you have completed the previous of **RemoteUnit** correctly. Also, assume that you have two additional functions already implanted correctly: **MoveBackward** and **StopMoving**.

Write a main function the does the following:

1. Define an array of remote control unit objects of size three, that will control three different cars.
2. For the first car, move it forward, open the front left window, stop the car, then close the opened window.
3. For the second car, move it backward.
4. For the third car, assume this car is facing an obstacle that you need to pass. Turn the wheels 45 degrees to the left. Move the car forward, turn the wheels 45 degrees to the right. Stop the car.
5. Count the number of cars that has frontLeft window open.
6. Write the code to Stop moving all cars.

```
int main() {
    RemoteUnit myarr[3];
    // -----
    myarr [0].MoveForward();
    myarr [0].OpenCloseFL(true);
    myarr [0].StopMoving();
    myarr [0].OpenCloseFL(false);
    // -----
    myarr [1].MoveBackward();
    // -----
    myarr [2].TurnWheels( true, 45);
    myarr [2].MoveForward();
    myarr [2].TurnWheels( false, 45);
    myarr [2].StopMoving();
    // -----
    int c=0;
    for(int i=0;i<3;i++) {
        if( myarr[i].isFLOpen() )
            c++;
    }
    cout << "number of cars with open frontLeft window = " << c << endl;
    // -----
    for(int i=0;i<3;i++)
        myarr[i].StopMoving();
    return 0;
}
```

Question 32 (30 points)

Trace the following programs and write the **generated output in the boxes below only?**

<pre>#include <iostream> using namespace std; class Emp { private: int id, age; public: Emp() { id = 1; age = 3; } int GetResult() { return age-2 ; } int SetAge(int value) { age = value; id = value + 2; return id + age; } }; int main() { Emp arr[5], e1; int res = 0; for (int i = 0; i < 5; i++) { res = arr[i].SetAge(i + res + e1.GetResult()); e1.SetAge(i); } for (int i = 0; i < 5; i++) cout << arr[i].GetResult() << endl; return 0; }</pre>	<pre>#include <iostream> using namespace std; class Test { private: int a; int b; public: Test (){ a = 32; b = 7; } void F2(int & x1, int y1){ Test u; u.SetVals(x1, y1); x1 += b; } void F3(int * p1, int & y1){ Test u; u.SetVals (*p1, y1); b = F88(); *p1 += b; } void Display(){ cout << a << endl << b << endl; } int F88() { return a + b; } void SetVals(int y, int o){ a = y; b = o; } }; int main() { int item1 = 2, item2 = 4; Test obj1, obj2; obj1.F2(item1, 3); obj1.F3(&item1, item2); obj2.F2(item2, 3); obj1.F3(&item2, item1); obj1.Display(); obj2.Display(); cout << item1 + item2 << endl; return 0; }</pre>
<p>Output:</p> <div><p>-1</p><p>1</p><p>7</p><p>21</p><p>51</p></div>	<p>Output:</p> <div><p>32</p><p>71</p><p>32</p><p>7</p><p>130</p></div>

--	--