



UNIVERSITY OF RUHUNA

Faculty of Engineering

End-Semester 2, Examination in Engineering - Model Answers, July
2022

Module No: EE2201 Module Name: Programming II

[1 hour and 30 minutes]

[Answer all questions. All questions carry equal marks]

Part II

Instructions for candidate

- It is recommended to answer in the order of questions Q1, Q2 and Q3.
 - Write all assumptions you make, if any.
-

Q1. Product of a certain business is represented by the class `cProduct` given in the Listing 1. Profit earned by selling the product is calculated as the difference between unit selling price and unit cost of production (`unitPrice - unitCost`) and it is returned by the member function `getUnitProfit()`. Give the definition of below listed member functions, by following the format given in the sample definition of the overridden default constructor in Listing 2.

- a) Overloaded Constructor `cProduct(double price, double cost)` [3 Marks]
- b) Getters [3 Marks]
- `double getUnitProfit()`
 - `double getUnitPrice()`
 - `double getUnitCost()`
- c) Setters [2 Marks]
- `void setUnitPrice(double uprice)`
 - `void setUnitCost(double ucost)`

Q2. A certain company employ temporary salesmen for selling products of the company. Each salesmen is paid a certain percentage of the profit of his/her sales. The class `cSalesman` that represents a salesman is given in Listing 3. Please note that the protected member `p` is of type `cProduct` defined in Listing 1.

a) Consider a member variable to the class `cSalesman` called `employeeNo`, which should be a unique number that can be assigned to each employee starting from 1 and can be increased by 1 at each declaration of an object of type `salesman`. What should be the access specifier and the type of this variable. [2 Marks]

b) Give the definition of overloaded constructor `cSalesman(..)` that sets the private member variables; `name` - name of the salesman, `p` - unit price and unit cost of the product, `commissionRate` - percentage of the profit paid to the salesman (commission), `nProductsSold` - number of products sold by the salesman, and the public member `employeeNo`. [2 Marks]

c) Give the definition of `getPayment()`, which should return the payment to the salesman, as a percentage of profit. [2 Marks]

d) Give the definition of overloaded assignment operator `cSalesman & operator=(const cSalesman & other)`. [2 Marks]

Q3. The incomplete declaration of class `cPermenentEmployee`, derived from the class `cSalesman` is given in Listing 4. It can represent a salesmen who is employed on permanent basis. The salary (payment) of a permanent salesman is calculated by adding the commission from the sales (as calculated in Listing 3) to a basic salary.

a) Complete the declaration of the class by adding necessary member functions; default constructor, overloaded constructor to initialize variables and destructor. [4 Marks]

b) Give the definition of [2 Marks]

- i. Overloaded constructor
- ii. the function `getPayment()` to return the salary.

c) What are the member variables of `cSalesman` that are accessible in `cPermenentSalesman` as private members. [2 Marks]

Listing 1: cProduct class declaration

```
class cProduct
{
    public:
        cProduct();
        cProduct(double price, double cost);
        virtual ~cProduct();
        double getUnitProfit(); double getUnitPrice();
        double getUnitCost(); void setUnitPrice(double uprice);
        void setUnitCost(double ucost);
    private:
        double unitPrice; double unitCost;
};
```

Listing 2: Default constructor of cProduct

```
cProduct::cProduct()
{
    unitCost = 0;
    unitPrice = 0;
}
```

Listing 3: cSalesman class declaration

```
class cSalesman
{
    public:
        cSalesman();
        cSalesman(..);
        virtual ~cSalesman();
        cSalesman(const cSalesman& other);
        cSalesman& operator=(const cSalesman& other);
        double getPayment();
    protected:
        string name;
        cProduct p;
        double nProductsSold;
        double commissionRate;
};
```

Listing 4: Incomplete cPermenentSalesman class declaration

```
class cPermenentSalesman : public cSalesman
{
    public:
        ..
        double getPayment();

    private:
        double basicSalary;
};
```