

## Task 2

### Robot Rental System

Any system has to consist of some users, and some service you are serving. In a robot rental system we have to have a robots and customers to rent the robots.

#### The program steps

At first, We have to make two classes, one for the robots, while the other is for the users. each class contains a private hidden variables, and a public methods to be able to use it. Private variables as shown in fig1. is to store the names, the IDs, the functions .. etc.

```
private:
    string name;
    string func;
    unsigned int id;
    unsigned int rent;
    unsigned int price;
    bool isrented;
    unsigned int day;
    unsigned int month;
    unsigned int year;
    unsigned int duration;
```

Fig1.

Public section has to contain the methods the user can use during using the program.

#### Methods to be created:

- 1- Adding a robot.
- 2- Adding a user.
- 3- Searching for a robot.
- 4- Searching for a user.
- 5- Displaying all users.
- 6- Displaying all robots.
- 7- Rent a robot to a user.

So, we have had to implement each of these methods.

Let's start with the first method, We had to receive the name of robot, the ID, the price per day, and the function, then the number of robots incremented by one as shown in fig2. In the second method we had to receive the name of user, the telephone number, and the e-mail address, then as usual incrementing the number of users by one as shown in fig3.

```

void Robot::add_robot()
{
    cout << "Enter the name: ";
    cin >> name;
    cout << "Enter the ID: ";
    cin >> id;
    cout << "Enter the price: ";
    cin >> price;
    cout << "Enter the function: ";
    cin >> func;
    Num++;
}

```

Fig2.

```

void User::add_user()
{
    cout << "Enter the name: ";
    cin >> name;
    cout << "Enter the phone number: ";
    cin >> phone;
    cout << "Enter the E-mail: ";
    cin >> mail;
    Num++;
}

```

Fig3.

The third method, we receive the robot name to display its information as shown in fig4.. The fourth method isn't differ much than the previous one, it's receiving the user's name to display his information as shown in fig5..

```

void Robot::robot_search(Robot R[])
{
    string name_;
    cout << "Enter the name: ";
    cin >> name_;
    for(int i=0; i<Num; i++)
        if(! (name_.compare(R[i].name)))
        {
            cout << "Name: " << R[i].name << endl;
            cout << "ID: " << R[i].id << endl;
            cout << "Price: " << R[i].price << endl;
            cout << "Function: " << R[i].func << endl;
            cout << "Days have been rented: " << R[i].rent << endl;
            if(R[i].isrented)
                cout << "Availability : Not available\n";
            else
                cout << "Availability : Available\n";
        }
}

```

Fig 4.

```

void User::user_search(User U[])
{
    string name_;
    cout << "Enter the name: ";
    cin >> name_;
    for(int i=0;i<Num;i++)
        if(! (name_.compare(U[i].name)))
        {
            cout << "Name: " << U[i].name << endl;
            cout << "Telephone: " << U[i].phone << endl;
            cout << "E-Mail: " << U[i].mail << endl;
            cout << "Rented robots: ";
            if(U[i].rent.size()==0)
                cout << "None";
            else
                for(int j=0;j<U[i].rent.size();j++)
                    cout << U[i].rent[j] << ",";
            cout << endl;
        }
}

```

Fig5.

The fifth method displays all the information for all users registered in the system. The sixth method is also displays all the information for all robot exist in the system.

The seventh method is displaying the users to choose which user is going to rent a robot, then it displays the robots and their information such as their availability and their functions and their price to choose which one to rent, The date you are going to rent it on, and how long you would rent the robot.

## Problems encountered

Due to using code blocks IDE, When running the program from multi-files, The error “Undefined reference” shows as fig6., So a file (Task2\_full.cpp) was attached as it all in one file to run the program.

|                  |  |
|------------------|--|
| C:\Users\Four... | undefined reference to `Robot::add_robot(std::__cxx11::basic_string<char, std::char_t... |
| C:\Users\Four... | undefined reference to `Robot::add_robot(std::__cxx11::basic_string<char, std::char_t... |
| C:\Users\Four... | undefined reference to `Robot::add_robot(std::__cxx11::basic_string<char, std::char_t... |
| C:\Users\Four... | undefined reference to `User::add_user(std::__cxx11::basic_string<char, std::char_tra... |
| C:\Users\Four... | undefined reference to `User::add_user(std::__cxx11::basic_string<char, std::char_tra... |

Fig6.