

KARADENİZ TECHNICAL UNIVERSITY – DEPARTMENT
of COMPUTER ENGINEERING

COM2009 Object Oriented Programming – Final Exam

09/01/2025 10:00

Exam Duration: 90 minutes

Student Number, Name, Surname and Signature:

Q1. The main() function below provides parts of a program that processes three different types of film information. This program contains three classes: **Film**, **DirectorCut**, and **ForeignFilm**. Create a class hierarchy that meets the following requirements using the main() function. (Write down the source code of the Film, DirectorCut and ForeignFilm classes, taking into account the inheritance hierarchy between them.) **NOTE:** The expected output of the code is as follows.

Film Class: Stores the movie title, director name, duration (in minutes), and quality (a value between 0 and 4) information, and uses the output function to print the information to the screen.

DirectorCut Class: Stores the revised duration (in minutes) and additional changes not in the original movie. The output() function prints both the base class information and this added information to the screen.

ForeignFilm Class: Stores the language information of the movie. The output() function prints both the base class information and the language information to the screen. - Aşağıdaki main() fonksiyonunda, üç farklı türde film bilgilerini işleyen bir programın parçaları verilmektedir. Bu program, **Film**, **DirectorCut** ve **ForeignFilm** olmak üzere üç adet sınıf içermektedir. main() fonksiyonunu kullanarak aşağıdaki gereksinimleri karşılayan bir sınıf hiyerarşisi oluşturunuz. (**Film**, **DirectorCut** ve **ForeignFilm** sınıflarını aralarındaki miras hiyerarşisini dikkate alarak yazınız) **NOT:** Kodun beklenen çıktısı aşağıda verildiği gibidir.

Film Sınıfı: Film başlığı, yönetmen adı, süre (dakika cinsinden) ve kalite (0-4 arası bir değer) bilgilerini saklamakta ve bilgileri ekrana yazdırmak için output fonksiyonu kullanmaktadır.

DirectorCut Sınıfı: Revize edilmiş süre (dakika cinsinden) ve orijinal filmde olmayan ek değişiklikler bilgilerini saklamaktadır. output() fonksiyonu, hem temel sınıfın bilgilerini hem de eklenen bu bilgileri ekrana yazdırır.

ForeignFilm Sınıfı: Filmin dili bilgisini saklamaktadır. output() fonksiyonu, hem temel sınıfın bilgilerini hem de dil bilgisini ekrana yazdırır.

```
int main() {
    Film f;
    f.store_title("Rear Window");
    f.store_director("Alfred Hitchcock");
    f.store_time(112);
    f.store_quality(4);
    cout << "Film -- \n";
    f.output();
    cout << "\n";
    DirectorCut d;
    d.store_title("Jail Bait");
    d.store_director("Ed Wood");
    d.store_time(70);
    d.store_quality(2);
    d.store_rev_time(72);
    d.store_changes("Extra footage not in original
included");
    cout << "DirectorCut -- \n";
    d.output();
    cout << "\n";
    ForeignFilm ff;
    ff.store_title("Jules and Jim");
```

```
ff.store_director("Francois Truffaut");
ff.store_time(104);
ff.store_quality(4);
ff.store_language("French");
cout << "ForeignFilm -- \n";
ff.output();
return 0;
}
```

Output

```
Film--
Title: Rear Window
Director: Alfred Hitchcock
Time: 112 mins
Quality: ****

DirectorCut--
Title: Jail Bait
Director: Ed Wood
Time: 70 mins
Quality: **
Revised time: 72 mins
Changes: Extra footage not in original included

ForeignFilm--
Title: Jules and Jim
Director: Francois Truffaut
Time: 104 mins
Quality: ****
Language: French
```

Answer:

continue on next page

Q2: Write down the output of the code blocks given below in a regular and meaningful way. - Aşağıda verilen kod bloklarının çıktılarını kurallı ve anlamlı bir şekilde veriniz.

(a) Write down the outputs according to the values 1, 2 and 3 entered by the user, in that order. - Kullanıcının girdiği 1, 2 ve 3 değerlerine göre çıktıları sırasıyla yazınız.

```
class TradesPerson {
public:
    void sayHi() { cout << "Just hi." << endl; }
};
class Tinker : public TradesPerson {
public:
    void sayHi() { cout << "Hi, I tinker." << endl; }
};
class Tailor : public TradesPerson {
public:
    void sayHi() { cout << "Hi, I tailor." << endl; }
};
int main() {
    TradesPerson* p;
    int which;
    // Prompt user for a number: 1=TradesPerson, 2=Tinker,
    3= Tailor
    do {
        cout << "1 == TradesPerson, 2 == Tinker, 3 == Tailor ";
        cin >> which;
    } while (which < 1 || which > 3);
    switch (which) {
        case 1: p = new TradesPerson; break;
        case 2: p = new Tinker; break;
        case 3: p = new Tailor; break;
    }
    p->sayHi();
    delete p;
    return 0; }
```

output if user prompt is 1

output if user prompt is 2

output if user prompt is 3

(b) Write down the screen output when the following code is run - Aşağıdaki kod çalıştırıldığında ekran çıktısı ne olur?

```
class StringManipulator {
private:
    string s;
public:
    StringManipulator(string str) { s = str; }
    void printOriginal() {
        cout << "Original string: " << s << endl;
    }
    void printReverse() {
        cout << "String in reverse: ";
        for (int i=s.length() - 1; i>=0; i -- ) {
            cout << s[i];
        }
        cout << endl;
    }
    void findCharacter(char c) {
        int index = - 1;
        for (int i=0; i < s.length(); i++) {
            if (s[i] == c) {
                index = i;
                break;
            }
        }
        if (index != - 1) {
            cout << "'" << c << "'is the " << (index+1) <<
            "th character in: " << s << endl;
        } else {
            cout << "'" << c << "' is not found in: " << s
            << endl;
        }
    }
};

int main() {
    StringManipulator sm("Karadeniz Teknik");
    sm.printOriginal();
    sm.printReverse();
    sm.findCharacter('n');
    return 0;
}
```

Output:

Q3. The following code contains an example of operator overloading. The necessary methods for increment, decrement and integer addition have been written. Fill in the blanks in the code accordance with the postfix and prefix rules and give the screen output after the main() function is run - Aşağıdaki kod operatör aşırı yükleme örneği içermektedir. artırma, azaltma ve tamsayı değerle toplama operatörleri için gerekli metodlar yazılmıştır. Postfix ve prefix kurallarına uygun olacak şekilde kodda boş olan yerleri doldurun ve main() fonksiyonu çalıştırdıktan sonraki ekran çıktısını verin.

```
class Counter {
private:
    int value;
public:
    Counter(int v = 0) : value(v) {}
    _____ operator++() {
        ++value; // Increment the value
        return *this; // Return the incremented value
    }
    Counter operator++(int) {
        _____
        _____
    }
    Counter& operator--() {
        _____
        _____
    }
    _____ operator--(int) {
        _____
        _____
        return temp; // Return the previous value
    }
    Counter operator+(_____ ) {
        _____
        _____
    }
    _____
    operator<<(_____ ) {
        os << "Value: " << c.value;
        return os;
    }
};

int main() {
    Counter c(5);
    cout << "Initial value:" << c << endl;
    cout << "Using postfix ++:" << c++ << endl;
    cout << "Using prefix ++:" << ++c << endl;
    c = c + 5;
    cout << "Using postfix ++:" << c-- << endl;
    cout << "Using prefix ++:" << --c << endl;
    return 0; }
```

Output:

