|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
|  | **Course:** | **Object Oriented Programming** | **Course Code:** | **CS-217** |
| **Program:** | **BS Computer Science** | **Semester:** | **Spring 2024** |
| **Duration:** | **40 minutes** | **Total Marks:** | **20** |
| **Date:** | **-** | **Weight:** | **-** |
| **Section:** | **D** | **Page(s):** | **1** |
| **Exam:** | **Quiz 4 (b)** | **Roll No.** |  |

**Question:  
  
a)**A university needs a system to manage different types of students. The students can be classified into three categories: UndergraduateStudent, GraduateStudent, and ExchangeStudent. All students must belong to one of these specialized types, and no generic student should be created. Each type of student is derived from a base class and has some shared and unique attributes and behaviors, as described below.

1. All Students have the following attributes in common: **name**, **id**, **gpa**, and a **calculateTuitionFee**() method.
2. Implement three classes: UndergraduateStudent, GraduateStudent, and ExchangeStudent, each inheriting from the Student class with unique attributes:
   * For UndergraduateStudent: **yearOfStudy** (int), **isFunded** (bool). They have a base tuition fee of $10000 and receive a discount based on their year of study (5% per year) and if they are currently being funded.
   * For GraduateStudent: **researchCredits** (int), **thesisFee** (double). They have an additional thesis fee and their tuition fee is calculated based on their research credits. ($100 per credit)
   * For ExchangeStudent: **homeUniversity** (string), **exchangeDuration** (int). They pay a fixed tuition fee based on the exchange duration. ($1000 per year)
3. Implement default and parameterized constructors, destructors, and a calculateTuitionFee() method in each derived class to calculate the tuition fee for the respective student type based on the provided  
   attributes.

**b)**Give output of the following main function:

int main() {

// Create student objects

Student\*\* students = new Student \* [3];

students[0] = new UndergraduateStudent("Alice", 1001, 3.8, 2, true);

students[1] = new GraduateStudent("Bob", 2002, 3.5, 12, 2000.0);

students[2] = new ExchangeStudent("Charlie", 3003, 4.0, "MIT", 1);

// Calculate and display tuition fees

for (int i = 0; i < 3; i++) {

cout << "Name: " << students[i]->name << endl;

cout << "ID: " << students[i]->id << endl;

cout << "GPA: " << students[i]->gpa << endl;

cout << "Tuition Fee: $" << students[i]->calculateTuitionFee() << endl;

cout << endl;

}

}

**SOLUTION:**

a)

// Base class Student

class Student {

protected:

string name;

int id;

double gpa;

public:

Student(string n, int i, double g) : name(n), id(i), gpa(g) {}

virtual double calculateTuitionFee() = 0; // Pure virtual function

};

// Derived class UndergraduateStudent

class UndergraduateStudent : public Student {

int yearOfStudy;

bool isFunded;

public:

UndergraduateStudent(string n, int i, double g, int y, bool f)

: Student(n, i, g), yearOfStudy(y), isFunded(f) {}

double calculateTuitionFee() {

double baseFee = 10000;

double discount = 0;

if (isFunded)

discount = 0.05 \* yearOfStudy;

return baseFee \* (1 - discount);

}

};

// Derived class GraduateStudent

class GraduateStudent : public Student {

int researchCredits;

double thesisFee;

public:

GraduateStudent(string n, int i, double g, int c, double f)

: Student(n, i, g), researchCredits(c), thesisFee(f) {}

double calculateTuitionFee() {

return 100 \* researchCredits + thesisFee;

}

};

// Derived class ExchangeStudent

class ExchangeStudent : public Student {

string homeUniversity;

int exchangeDuration;

public:

ExchangeStudent(string n, int i, double g, string h, int d)

: Student(n, i, g), homeUniversity(h), exchangeDuration(d) {}

double calculateTuitionFee() {

return 1000 \* exchangeDuration;

}

};

**b)**

**Name: Alice**

**ID: 1001**

**GPA: 3.8**

**Tuition Fee: $9000**

**Name: Bob**

**ID: 2002**

**GPA: 3.5**

**Tuition Fee: $3200**

**Name: Charlie**

**ID: 3003**

**GPA: 4**

**Tuition Fee: $1000**