

KIỂM TRA - Lập trình hướng đối tượng

Câu 1: Hãy định nghĩa class **Person** với các thuộc tính: *Họ tên, ngày sinh, quê quán, giới tính*. Các phương thức khởi tạo, get và set cho các thuộc tính, nhập và hiển thị thông tin.

Câu 2: Định nghĩa class **Student** kế thừa từ **Person**, có thêm các thuộc tính: *mã sinh viên, email, điểm trung bình, tổng số tín chỉ tích lũy*. Các phương thức khởi tạo, get và set cho các thuộc tính, nhập và hiển thị thông tin.

Câu 3: Định nghĩa lớp **StudentList** với các yêu cầu sau:

- Sử dụng ArrayList để lưu trữ
- Thực hiện các yêu cầu: nhập danh sách Student, In lại danh sách đã nhập, cho biết tên Student có điểm trung bình cao nhất, lưu toàn bộ danh sách vào file “sinhvien.dat”

Câu 4: Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class ThongDiep{
    public ThongDiep ()
    {    System.out.println("Xin chao");}
    public void sayGoodbye()
    {    System.out.println("Tam biet");}
}
class Messenger extends ThongDiep {
    public Messenger (){
        super();
        System.out.println("Xin Chao Java");
    }
    public void sayGoodbye()
    {    System.out.println("Tam biet Java");}
}
public class Demo {
    public static void main(String args[]){
        Messenger m1 = new Messenger ();
        m1.sayGoodbye();    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1: Hãy định nghĩa class **Người** với các thuộc tính: *Họ tên, ngày sinh, địa chỉ, giới tính*. Các phương thức khởi tạo, get và set cho các thuộc tính, nhập và hiển thị thông tin.

Câu 2: Định nghĩa class **SinhVien** kế thừa từ **Người**, có thêm các thuộc tính: *mã sinh viên, điện thoại, điểm trung bình, tổng số tín chỉ còn nợ*. Các phương thức khởi tạo, get và set cho các thuộc tính, nhập và hiển thị thông tin.

Câu 3: Định nghĩa lớp **DSSinhvien** với các yêu cầu sau:

- Sử dụng ArrayList để lưu trữ
- Thực hiện các yêu cầu: nhập danh sách **SinhVien**, In lại danh sách đã nhập, cho biết tên **SinhVien** có điểm trung bình thấp nhất, lưu toàn bộ danh sách vào file “student.dat”

Câu 4: Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class ThongDiep{
    public ThongDiep ()
    {    System.out.println("Xin chào");}
    public void sayGoodbye()
    {    System.out.println("Tam biệt");}
}

class Messenger extends ThongDiep {
    public Messenger (){
        super();
        System.out.println("Xin Chào Java");
    }
    public void sayGoodbye()
    {    System.out.println("Tam biệt Java");}
}

public class Demo {
    public static void main(String args[]){
        Messenger m2 = new Messenger ();
        m2.sayGoodbye();    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1: Hãy định nghĩa class **Người** với các thuộc tính: *Họ tên, ngày sinh, địa chỉ, giới tính*. Các phương thức khởi tạo, get và set cho các thuộc tính, nhập và hiển thị thông tin.

Câu 2: Định nghĩa class **NhanVien** kế thừa từ **Người**, có thêm các thuộc tính: *phòng ban, hệ số lương, hệ số phụ cấp*. Các phương thức khởi tạo, get và set cho các thuộc tính, tính lương thực lĩnh, nhập và hiển thị thông tin.

*Trong đó: lương thực lĩnh = Lương cơ bản * (hệ số lương + hệ số phụ cấp)*

Câu 3: Định nghĩa lớp **NhanVienList** với các yêu cầu sau:

- Sử dụng ArrayList để lưu trữ
- Thực hiện các yêu cầu: nhập danh sách **NhanVien**, In lại danh sách đã nhập, cho biết tên **NhanVien** có điểm thực lĩnh thấp nhất, lưu toàn bộ danh sách vào file “**nhanvien.dat**”

Câu 4: Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class ThongDiep{
    public ThongDiep ()
    {    System.out.println("Xin chào");}
    public void sayGoodbye()
    {    System.out.println("Tạm biệt");}
}
class Messenger extends ThongDiep {
    public Messenger (){
        super();
        System.out.println("Xin Chào Java");
    }
    public void sayGoodbye()
    {    System.out.println("Tạm biệt Java");}
}
public class Demo {
    public static void main(String args[]){
        ThongDiep m1 = new ThongDiep ();
        M1.sayGoodbye();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1: Hãy định nghĩa class **Person** với các thuộc tính: *Họ tên, ngày sinh, địa chỉ, giới tính*. Các phương thức khởi tạo, get và set cho các thuộc tính, nhập và hiển thị thông tin.

Câu 2: Định nghĩa class **Employee** kế thừa từ **Person**, có thêm các thuộc tính: *hệ số lương, lương tăng thêm, vị trí công việc*. Các phương thức khởi tạo, get và set cho các thuộc tính, tính lương thực lĩnh, nhập và hiển thị thông tin.

*Trong đó: lương thực lĩnh = Lương cơ bản * hệ số lương + lương tăng thêm*

Câu 3: Định nghĩa lớp **EmployeeList** với các yêu cầu sau:

- Sử dụng ArrayList để lưu trữ
- Thực hiện các yêu cầu: nhập danh sách **Employee**, In lại danh sách đã nhập, cho biết tên **Employee** có điểm thực lĩnh cao nhất, lưu toàn bộ danh sách vào file “**nhanvien.dat**”

Câu 4: Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class ThongDiep{
    public ThongDiep ()
    {    System.out.println("Xin chào");}
    public void sayGoodbye()
    {    System.out.println("Tạm biệt");}
}
class Messenger extends ThongDiep {
    public Messenger (){
        super();
        System.out.println("Xin Chào Java");
    }
    public void sayGoodbye()
    {    System.out.println("Tạm biệt Java");}
}
public class Demo {
    public static void main(String args[]){
        ThongDiep m1 = new ThongDiep ();
        M1.sayGoodbye();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1-01

- Hãy định nghĩa class Person với các thuộc tính: Họ tên, ngày sinh, địa chỉ, giới tính và các phương thức khởi tạo, hiển thị thông tin.
- Định nghĩa class Student kế thừa từ Person, có thêm các thuộc tính: mã sinh viên, email, điểm tổng kết và các phương thức khởi tạo, hiển thị thông tin.

Câu 2-01

- Tạo form cho phép nhập các thông tin sinh viên như định nghĩa ở trên.
- Khi người dùng click nút Thêm sinh viên thì chương trình khởi tạo đối tượng sinh viên với các giá trị thuộc tính lấy từ form người dùng nhập và hiển thị thông tin lên màn hình console.
- Khi người dùng click nút Lưu dữ liệu thì ghi toàn bộ thông tin vào file sinhvien.dat

Câu 3-01

Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class Msg{
    public Msg()
    {    System.out.println("Hello");}
    public void sayGoodbye()
    {    System.out.println("Goodbye");}
}
class TestMsg extends Msg{
    public TestMsg(){
        super();
        System.out.println("Hello Java");
    }
    public void sayGoodbye()
    {    System.out.println("Goodbye Java");}
}
public class Demo {
    public static void main(String args[]){
        Msg m1 = new Msg();
        Msg m2 = new TestMsg();
        m1.sayGoodbye();
        m2.sayGoodbye();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1-02

- Hãy định nghĩa class Person với các thuộc tính: Họ tên, ngày sinh, địa chỉ, giới tính và các phương thức khởi tạo, hiển thị thông tin.
- Định nghĩa class NhanVien kế thừa từ Person, có thêm các thuộc tính: phòng ban (nhận 1 trong các giá trị: Thiết bị, nhân sự, R&D, Kinh doanh), hệ số lương, thâm niên, lương cơ bản và các phương thức khởi tạo, tính lương thực lĩnh, hiển thị thông tin cá nhân

*Trong đó: lương thực lĩnh = Lương cơ bản * hệ số (1 + thâm niên/100)*

Câu 2-02

- Tạo form cho phép nhập các thông tin nhân viên như định nghĩa ở trên.
- Khi người dùng click nút Thêm nhân viên thì chương trình khởi tạo đối tượng nhân viên với các giá trị thuộc tính lấy từ form người dùng nhập và hiển thị thông tin lên màn hình console.
- Khi người dùng click nút Lưu dữ liệu thì ghi toàn bộ thông tin vào file nhanvien.dat

Câu 3-02

Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class Msg{
    public Msg(){
        System.out.println("Hello");
    }
    public void sayGoodbye(){
        System.out.println("Goodbye");
    }
}
class TestMsg extends Msg{
    public void sayGoodbye(){
        System.out.println("Goodbye Java");
    }
}
public class Demo {
    public static void main(String args[]){
        Msg m1 = new Msg();
        Msg m2 = new TestMsg();
        m1.sayGoodbye();
        m2.sayGoodbye();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1-03

- Hãy định nghĩa class Person với các thuộc tính: Họ tên, ngày sinh, địa chỉ, giới tính và các phương thức khởi tạo, hiển thị thông tin.
- Định nghĩa class NhanVien kế thừa từ Person, có thêm các thuộc tính: phòng ban, hệ số lương, thâm niên, lương cơ bản và các phương thức khởi tạo, tính lương thực lĩnh, hiển thị thông tin cá nhân

*Trong đó: lương thực lĩnh = Lương cơ bản * hệ số (1 + thâm niên/100)*

Câu 2-03

- Hãy đọc file nhanvien.dat gồm danh sách các nhân viên với định dạng lưu trữ như sau: mỗi nhân viên được lưu trên 1 dòng, mỗi thuộc tính phân cách bởi ký tự '\$'. Dữ liệu đọc được lưu dưới dạng một mảng các đối tượng nhân viên (arraylist)
- Hiển thị danh sách trên trong một table (hiển thị trên console chỉ được 1.5 điểm).

Câu 3-03

Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
class Msg{
    public Msg(){
    public Msg(String str)
    {    System.out.println("Hello" + str); }
    public void sayGoodbye()
    {    System.out.println("Goodbye"); }
}
class TestMsg extends Msg{
    public TestMsg(String str){
        System.out.println("Hello" + str);}
    public void sayGoodbye(){
        System.out.println("Goodbye Java");}
}
public class Demo {
    public static void main(String args[]){
        Msg m1 = new Msg("Mark");
        Msg m2 = new TestMsg("Java");
        m1.sayGoodbye();
        m2.sayGoodbye();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1-04

- Hãy định nghĩa class Person với các thuộc tính: Họ tên, ngày sinh, địa chỉ, giới tính và các phương thức khởi tạo, hiển thị thông tin.
- Định nghĩa class Student kế thừa từ Person, có thêm các thuộc tính: mã sinh viên, email, điểm tổng kết và các phương thức khởi tạo, hiển thị thông tin.

Câu 2-04

- Hãy đọc file sinhvien.dat gồm danh sách các Sinh viên với định dạng lưu trữ như sau: mỗi Sinh viên được lưu trên 1 dòng, mỗi thuộc tính phân cách bởi ký tự '\$'. Dữ liệu đọc được lưu dưới dạng một mảng các đối tượng Sinh viên (arraylist).
- Hiển thị danh sách trên trong một table (hiển thị trên console chỉ được 1.5 điểm).

Câu 3-04

Hãy chỉ ra lỗi sai của đoạn chương trình sau:

```
class Msg{
    public Msg(){
        System.out.println("Hello");
    }
    public void sayGoodbye(){
        System.out.println("Goodbye");
    }
}
class TestMsg extends Msg{
    public void sayGoodbye(){
        System.out.println("Goodbye Java");
    }
}
public class Demo {
    public static void main(String args[]){
        Msg m1 = new Msg("OOP");
        TestMsg m2 = new Msg();
        m1.sayGoodbye();
        m2.sayGoodbye();
    }
}
```


KIỂM TRA - Lập trình hướng đối tượng

Câu 1-05

- Hãy định nghĩa class **Sinhvien** với các thuộc tính: mã sinh viên, họ tên, ngày sinh, giới tính, điểm trung bình và các phương thức khởi tạo, hiển thị thông tin.
- Định nghĩa class **SinhvienATTT** kế thừa từ **Sinhvien**, có thêm thuộc tính: học phí và phương thức khởi tạo, hiển thị thông tin.
- Định nghĩa class **SinhvienMatMa** kế thừa từ **Sinhvien**, có thêm thuộc tính: đơn vị, lương và phương thức khởi tạo, hiển thị thông tin.

Câu 2-05

Hãy tạo các form cho phép nhập dữ liệu sinh viên, sinh viên mật mã thì lưu dữ liệu vào file svmm.dat, sinh viên an toàn thì lưu dữ liệu vào file svat.dat

Câu 3-05

Hãy chỉ ra lỗi sai của đoạn chương trình sau:

```
interface Action{
    public void drive(){};
    public void stop(){};
}
class Vehicle implements Action{
    private int id;
    public void Vehicle(int id){
        this.id = id;
        System.out.println("Hello world");
    }
}
public class Demo {
    public static void main(String args[]){
        Vehicle m = new Vehicle ();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1-06

Hãy định nghĩa class Word với các thuộc tính: id, en(từ tiếng anh), vn(nghĩa tiếng việt) và các phương thức khởi tạo, hiển thị thông tin, các phương thức getters/setters.

Câu 2-06

- Khởi tạo một danh sách (> 10 từ), có thể tự khởi tạo hoặc nhập dữ liệu
- Tạo một Form tìm kiếm, thực hiện tìm kiếm một từ trong từ điển (sử dụng thuật toán tìm nhị phân, tìm tuần tự được $\frac{1}{2}$ số điểm). Nếu tìm thấy thì hiển thị nghĩa tiếng việt, trái lại thực hiện theo câu 2.c
- Bổ sung một từ vào từ điển nếu từ đó chưa có trong từ điển

Chú ý: các từ trong danh sách phải đúng thứ tự từ điển (theo từ tiếng anh)

Câu 3-06

Hãy chỉ ra lỗi sai của đoạn chương trình sau:

```
interface Action{
    public void drive();
    public void stop();
}
class Vehicle implements Action{
    private int id;
    public void Vehicle(int id){
        this.id = id;
        System.out.println("Hello world");
    }
    public void drive(){
        System.out.println("I'm driving");
    }
}
public class Demo {
    public static void main(String args[]){
        Vehicle m = new Vehicle ();
        m.drive();
        m.stop();
    }
}
```

KIỂM TRA - Lập trình hướng đối tượng

Câu 1-07 (4 điểm)

- (2 điểm) Hãy định nghĩa class Thí sinh với các thuộc tính: họ tên, ngày sinh, địa chỉ và các phương thức khởi tạo, nhập và in thông tin
- (1 điểm) Định nghĩa class Thí sinh khối A kế thừa từ Thí sinh, có thêm các thuộc tính: điểm toán, lý, hóa và các phương thức khởi tạo, nhập, in thông tin
- (1 điểm) Định nghĩa class Thí sinh khối C kế thừa từ Thí sinh, có thêm các thuộc tính: điểm văn, sử, địa và các phương thức khởi tạo, nhập, in thông tin.

Câu 2-07 (4 điểm)

Thực hiện nhập danh sách n thí sinh của cả 2 khối A, C và in danh sách thí sinh trúng tuyển (tổng điểm > 20) ra màn hình.

Câu 3-07 (2 điểm)

Hãy cho biết kết quả (và lý do) của đoạn chương trình sau:

```
public class FileDemo{
    public static void method1(File f1) {
        String contents[] = f1.list();
        for (int i=0; i< contents.length; i++){
            System.out.println(contents[i] + "\n");
        }
    }
    public static void main(String[] args){
        method1 (new File("D:\\Documents\\act"));
    }
}
```