



Bộ môn Công nghệ Phần mềm
Viện CNTT & TT
Trường Đại học Bách Khoa Hà Nội

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Các bài tập từ bài 01 – 05



Câu 1

■ Lựa chọn nào sau đây là tên biến không hợp lệ trong Java?

1. \$char

2. 1MyNumber ✓

3. case ✓

4. _int



Câu 2

- Câu lệnh nào dưới đây không sinh ra lỗi biên dịch?
 1. `char my_char = 'c';` ✓
 2. `char your_char = 'int';`
 3. `char what = 'Hello';`
 4. `char what_char = "L";`

Câu 3

- Cho đoạn mã nguồn dưới đây:

```
public class Unary{  
    public static void main(String[] args) {  
        int x = 7;  
        int y = 6*x++;  
        System.out.println (" y= " + y);  
        int a = 7;  
        int b = 6*++a;  
        System.out.println (" b= " + b);  
    }  
}
```

- Kết quả nhận được khi thực thi đoạn mã này?

1. y= 48 b= 48
2. y= 42 b= 48
3. y= 48 b= 42
4. y= 42 b= 42





Câu 4

- Cho đoạn mã nguồn dưới đây:

```
byte a = 7;
```

```
short b = 3;
```

```
c = a * ++b;
```

- Giả sử đoạn mã hoạt động chính xác, hãy cho biết kiểu dữ liệu có thể của biến c?
 1. short, int, long, float, double
 2. short, char, int, float, double
 3. byte, short, int, long, float, double
 4. int, long, float, double ✓



Câu 5

- Cho đoạn mã nguồn dưới đây:

```
int x = 9;
```

```
int y = -2;
```

```
System.out.println("output: " + x%y);
```

- Kết quả nhận được khi thực thi đoạn mã này?

1. output: -1

2. output: 1

3. output: 4.5

4. output: 4



Câu 6

- Cho biết kết quả thực hiện của đoạn mã sau?

```
public class LastLaugh {  
    . . . public static void main(String args[]) {  
    . . .     System.out.print("H" + "a");  
    . . .     System.out.print('H' + 'a');  
    . . . }  
}
```

Kết quả: hiển thị trên màn hình
Ha169

Câu 7

- Cho biết kết quả thực hiện của đoạn mã sau?

```
public class MyClass {  
    . . . public static void main(String[] args) {  
    . . .     System.out.println(1 + 2 + "3");  
    . . .     System.out.println("1" + 2 + 3);  
    . . . }  
}
```

Kết quả: hiển thị trên màn hình

33

123

Câu 8

- Viết chương trình tính **HarmonicSum** $Harmonic(n) = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$
- Đoạn chương trình chưa hoàn thiện, hãy bổ sung những chỗ còn trống:

```
public 1 HarmonicSum {  
    public static 2 main (String[] args) {  
        int maxDenominator = 50000;  
        3 sum = 0.0;  
        for (int denominator = 1; denominator <= maxDenominator; ++denominator) {  
            4  
        }  
        5.out.println("The sum is: " + sum);  
    }  
}
```

1. class
2. void
3. double
4. sum += (double) 1/denominator;
5. System



Câu 9

- Trong đoạn mã này giá trị của X là bao nhiêu để in ra tất cả các phần tử trong mảng?

```
int values[] = {1, 2, 3, 4, 5, 6, 7, 8};  
for(int i=0; i< X; ++i)  
System.out.println(values[i]);
```

Giá trị của X là:
7

Câu 10

- Cho biết kết quả khi thực thi đoạn chương trình sau:

```
int x = 3;  
int y = 8;  
System.out.println(y / x);  
System.out.println((double)y / x);  
System.out.println((double)(y / x));  
System.out.println(y / (double)x);
```

Kết quả: hiển thị trên màn hình

2

2.6666666666666665

2.0

2.6666666666666665



Câu 11

- Chú trọng đến các tính chất quan trọng trong khi bỏ qua các chi tiết ít quan trọng được gọi là:
 1. Trừu tượng hóa ✓
 2. Đa hình
 3. Đóng gói
 4. Che giấu thông tin



Câu 12

- "Che đi các chi tiết cài đặt và chỉ cho thấy giao diện của mô-đun" là đặc điểm của khái niệm nào?
 1. Trừu tượng hóa
 2. Đa hình
 3. Đóng gói ✓
 4. Tái sử dụng



Câu 13

■ Phương án nào là khai báo phương thức hợp lệ?

1. `void method1 { /* ... */ }`
2. `void method2() { /* ... */ }` ✓
3. `void method3(void) { /* ... */ }`
4. `method4() { /* ... */ }`
5. `method5(void) { /* ... */ }`



Câu 14

- Cho định nghĩa lớp như sau, thuộc tính nào có thể truy nhập bên ngoài gói my.project:

```
package my.project;  
public class MyClass {  
    int i;  
    public int j;  
    private int l;  
}
```

Biến j



Câu 15

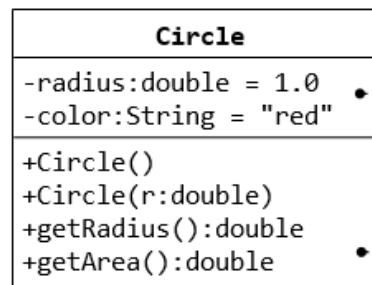
- Cho biết đoạn mã sau biên dịch có thành công không?

```
import java.util.*;  
package com.cnpm.hust;  
public class Test {  
    public static void main(String[] args) {  
        System.out.println("Hello world");  
    }  
}
```

Đoạn mã có lỗi biên dịch

Câu lệnh khai báo package phải để trên đầu tiên trong file mã nguồn

Câu 16



\pm varName:type = default-value

- denotes private access
+ denotes public access

\pm methodName(parmName:type,...):retunType

- Đoạn mã xây dựng lớp Circle theo sơ đồ lớp UML tương ứng, tuy nhiên còn một số chỗ trống chưa hoàn thành, hãy điền vào chỗ trống các nội dung thích hợp?

```
public A Circle {  
    B double radius;  
    private C color;  
    D Circle() {  
        radius = 1.0;  
        color = "red";  
    }  
    public E (double r) {  
        radius = r;  
        color = "red";  
    }  
}
```

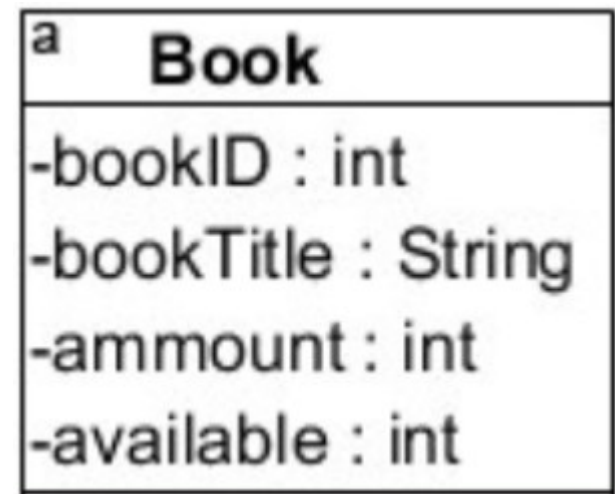
```
public double F {  
    return radius;  
}  
public double getArea() {  
    return G;  
}  
}
```

A. class
B. private
C. String
D. public

E. Circle
F. getRadius()
G. radius*radius*3.14

Câu 17

- Viết chương trình bằng ngôn ngữ Java xây dựng lớp Book như sơ đồ lớp UML sau, viết các phương thức get/set cho các thuộc tính của lớp?





Câu 18

- Từ khóa new dùng để làm gì? Giải thích chuyện xảy ra khi dùng từ khóa này trong một ứng dụng Java?



Câu 19

- Hàm khởi tạo mặc định là gì? Các thuộc tính của một đối tượng được khởi tạo như thế nào nếu lớp đó không có hàm khởi tạo nào do lập trình viên viết.



Câu 20

- a) Tạo một lớp biểu diễn điểm tọa độ nguyên trong mặt phẳng (Point) gồm các thuộc tính: Tên điểm (Chuỗi ký tự ví dụ: "A", "BB", "C"...), Hoành độ (số nguyên), Tung độ (số nguyên).
 - Viết các phương thức get/set cho lớp Point
 - Viết các phương thức khởi tạo sau:
 - + Phương thức khởi tạo không có tham số nào: Gán cho hoành độ và tung độ = 0, tên điểm mặc định là "A".
 - + Phương thức khởi tạo nhận hai số nguyên làm tham số
 - + Phương thức khởi tạo nhận 1 xâu ký tự và 2 số nguyên làm tham số
 - Viết phương thức nhập vào một điểm từ bàn phím: Nhập tung độ và hoành độ từ bàn phím.
 - Viết phương thức in thông tin một điểm ra màn hình dưới dạng: Tên_điểm(Tung_độ, Hoành_độ). Ví dụ: A(2, 4).
 - Viết phương thức so sánh hai đối tượng: hai đối tượng là bằng nhau nếu chúng có cùng tọa độ (tung độ bằng nhau và hoành độ bằng nhau)
- b) Viết hàm main tạo các đối tượng của lớp Point đã được định nghĩa trong câu trước và thực hiện lời gọi các phương thức của các đối tượng đó.

Câu 21

- Cho biết kết quả thực hiện của đoạn mã sau? Giải thích ngắn gọn?

```
public class Test {  
    public static void main(String[] args) {  
        String s = "ab" + "12";  
        String t = "ab" + 12;  
        String u = new String("ab12");  
        System.out.println((s==t) + " " + (s==u));  
    }  
}
```

Kết quả thực hiện đoạn chương trình:
Giải thích:

Câu 22

- Cho biết kết quả thực hiện của đoạn mã sau? Giải thích ngắn gọn?

```
public class ParameterPass {  
    public static void main(String[] args) {  
        int i = 0;  
        addTwo(i++);  
        System.out.println(i);  
    }  
    static void addTwo(int i) { i += 2; }  
}
```

Kết quả thực hiện đoạn chương trình:
Giải thích:

Câu 23

- Cho đoạn chương trình dưới đây chưa hoàn thiện, hãy bổ sung những chỗ còn trống:

```
class Student {  
    1 name;  
    int rollNo;  
    static String collegeName;  
    2 int counter = 0;  
    public Student(String name) {  
        this.name = name;  
        this.rollNo = setRollNo();  
    }  
    static void setCollege(String name) {  
        collegeName = name;  
    }  
    static int setRollNo() {  
        3  
        return counter;  
    }  
    void 4 {  
        System.out.println("name: " + this.name);  
        System.out.println("rollNo: " + this.rollNo);  
    }  
}
```

Output:

```
name : Alice  
rollNo : 1  
  
name : Bob  
rollNo : 2
```

```
public class Test {  
    public static void main(String[] args) {  
        5.setCollege("ĐH Bach Khoa");  
        Student s1 = new Student("Alice");  
        Student s2 = new Student("Bob");  
        s1.getStudentInfo();  
        s2.getStudentInfo();  
    }  
}
```


Câu 24

- Kết quả khi biên dịch và thực thi đoạn chương trình dưới đây?

```
class Cau3 {  
    static int m(int i) {  
        System.out.print(i + ", ");  
        return i;  
    }  
    public static void main(String s[]) {  
        m(m(1) + m(2) % m(3) * m(4));  
    }  
}
```

Kết quả thực hiện đoạn chương trình:
Giải thích:



Câu 25

- Trong một lớp, người ta xây dựng hai dạng nạp chồng (overload) cho một phương thức. Một dạng được định nghĩa là phương thức tĩnh (static) và một dạng khác được định nghĩa là phương thức thông thường (non-static). Cách thực hiện overload này có chính xác không? Giải thích ngắn gọn.

Trả lời:
Giải thích:

Câu 26

- Trong lớp dưới đây, “method” đang được nạp chồng hay bị trùng lặp (lỗi)? Giải thích ngắn gọn.

```
public class MainClass {  
    . . . void method(int . . . a) {  
        . . . System.out.println(1);  
    }  
    . . . void method(int[] a) {  
        . . . System.out.println(2);  
    }  
}
```

Trả lời:
Giải thích:

Câu 27

- Kết quả khi biên dịch và thực thi đoạn chương trình dưới đây?

```
class A {  
    ... public A(int i) {  
        ... System.out.println(1);  
    }  
    ... public A() {  
        ... this(10);  
        ... System.out.println(2);  
    }  
    ... void A() {  
        ... A(10);  
        ... System.out.println(3);  
    }  
    ... void A(int i) {  
        ... System.out.println(4);  
    }  
}  
public class MainClass {  
    ... public static void main(String[] args) {  
        ... new A().A();  
    }  
}
```

Trả lời:
Giải thích:

Câu 28

- Kết quả khi biên dịch và thực thi đoạn chương trình dưới đây?

```
class ClassOne {  
    void method(String s1) {  
        method(s1, s1+s1);  
    }  
    void method(String s1, String s2) {  
        method(s1, s2, s1+s2);  
    }  
    void method(String s1, String s2, String s3) {  
        System.out.println(s1+s2+s3);  
    }  
}  
  
public class MainClass {  
    public static void main(String[] args) {  
        ClassOne one = new ClassOne();  
        one.method("JAVA");  
    }  
}
```

Trả lời:
Giải thích:



Bài tập tổng hợp

- **Yêu cầu bài tập:** Xây dựng một chương trình đơn giản giúp bác tổ trưởng tổ dân phố quản lý thông tin các cuốn sách trong một thư viện (tủ sách nhỏ) của tổ dân phố 7, phường Bách Khoa. Chương trình có giao diện dòng lệnh (console application), khi người dùng muốn thực hiện một chức năng thì sẽ nhập vào số thứ tự của mục chức năng tương ứng → nhấn Enter và chương trình sẽ thực thi.
- Các chức năng chính của chương trình gồm có:
 1. Xem thông tin của thư viện → hiển thị thông tin thư viện như: tên thư viện, số đầu sách hiện có trong thư viện và danh sách tiêu đề các cuốn sách
 2. Thêm đầu sách mới → thêm một đầu sách mới vào thư viện
 3. Tìm kiếm sách → tìm kiếm thông tin đầu sách trong thư viện theo ID đầu sách
 4. Mượn một cuốn sách → khi có người đến mượn một cuốn sách, bác tổ trưởng sẽ dùng chức năng này để cập nhật số lượng sách còn lại của đầu sách tương ứng
 5. Trả một cuốn sách → khi có người đến trả một cuốn sách, chức năng này giúp cập nhật số lượng sách của đầu sách trong thư viện
 6. Kết thúc và thoát chương trình



Bài tập tổng hợp

- Giao diện chương trình có dạng như sau:

```
===== Main menu =====  
1. Show library information  
2. Add new book  
3. Find book  
4. Borrow a book  
5. Return a book  
6. Exit  
  
===== Enter menu ID (1-6) :
```

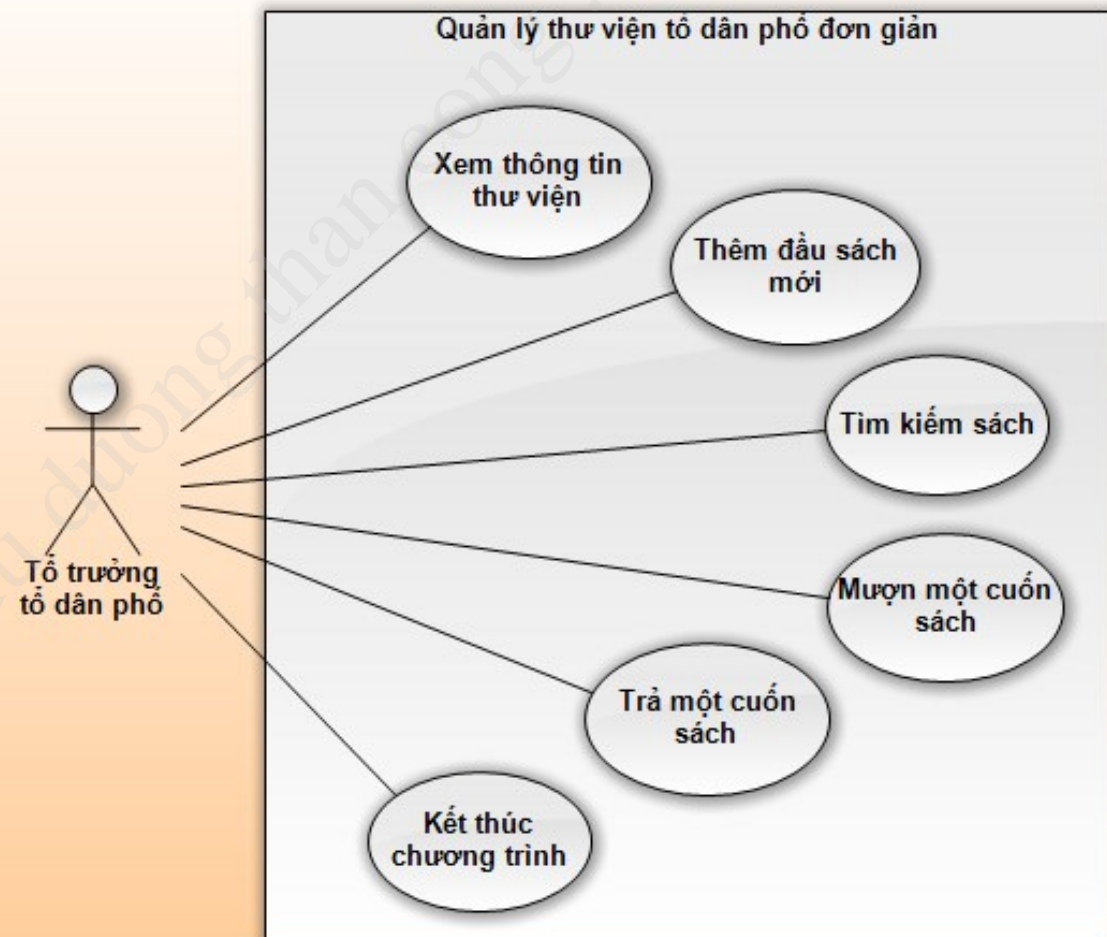


Bài tập tổng hợp

- **1. Xây dựng biểu đồ usecase cho bài toán trên?**

Bài tập tổng hợp

■ 1. Xây dựng biểu đồ usecase cho bài toán trên?





Bài tập tổng hợp

- **2. Xây dựng mã nguồn chương trình quản lý thư viện đơn giản trên?**



Bài tập tổng hợp

- **Phân tích:** để giải quyết bài toán này chúng ta xây dựng 3 lớp
 - Lớp **Book** đóng gói các thông tin về đầu sách trong thư viện, mỗi đối tượng của lớp Book là một đầu sách trong thư viện.
 - Lớp **Library** chứa các thông tin chung về thư viện như: tên thư viện,... và danh sách các đầu sách mà người dân trong tổ dân phố có thể mượn trong thư viện, để thể hiện điều này, đơn giản là chúng ta xây dựng một mảng (array) các đối tượng Book trong lớp Library. Các chức năng chính (thêm sách, tìm sách, mượn sách, trả sách) sẽ được thực hiện trong lớp Library này.
 - Lớp **Program** chứa phương thức **main()** để thực thi chương trình. Lớp Program cũng chịu trách nhiệm giao tiếp với người sử dụng chương trình (hiển thị menu chương trình, đọc lựa chọn người dùng nhập vào, thực thi chức năng qua lời gọi phương thức của đối tượng Library).

Bài tập tổng hợp

■ **Thiết kế** sơ đồ lớp

