



# BÀI TẬP

## CHƯƠNG TRÌNH CHỨNG CHỈ LẬP TRÌNH JAVA & J2EE

Môn học:

**NHẬP MÔN LẬP TRÌNH  
VỚI JAVA - JPCB**





# BÀI 1: Giới thiệu về lập trình Java



Giúp học viên biết cách cấu hình, biên dịch và thực thi ứng dụng Java sử dụng JDK. Làm quen với cách viết chương trình Java

## 1.1. Bài 1

Cài đặt và cấu hình Java

## 1.2. Bài 2

Viết ứng dụng xuất ra màn hình lời chào “Hello World!!!”

## 1.3. Bài 3:

Viết ứng dụng cho phép nhập vào tên người và xuất ra lời chào

Ví dụ:

Nhập họ tên: Văn Phú Cường

Xin chào Văn Phú Cường

## 1.4. Bài 4

Viết ứng dụng cho phép nhập vào hai số nguyên và in ra tổng của chúng

Ví dụ:

Nhập số thứ nhất: 5

Nhập số thứ hai: 10

Kết quả:  $5 + 10 = 15$



## BÀI 2: Ngôn ngữ Java



*Giúp học viên nắm các khái niệm cơ bản trong Java như đối tượng, lớp, kiểu dữ liệu, chuyển kiểu dữ liệu, biến và hằng, các toán tử, các cấu trúc điều khiển.*

### 2.1. Bài 1:

Có gì sai trong đoạn lệnh sau? Hãy cho biết cách khắc phục lỗi trên để cộng một vào tổng của  $x + y$

```
System.out.println( ++(x + y) );
```

### 2.2. Bài 2:

Lập trình cho phép nhập vào từ bàn phím độ dài các cạnh  $a, b, c$  của tam giác. Sau đó tính diện tích tam giác và xuất kết quả ra màn hình

### 2.3. Bài 3:

Lập trình cho phép nhập vào từ bàn phím chiều dài, chiều rộng của một hình chữ nhật. Thực hiện các yêu cầu sau:

- a) Tính chu vi
- b) Tính diện tích

Sau đó xuất các kết quả ra màn hình

### 2.4. Bài 4:

Lập trình cho phép nhập vào từ bàn phím số nguyên dương có 4 ký số. Sau đó tính và xuất ra màn hình số nút của số vừa nhập.

Ví dụ:  $\text{num} = 2873 \rightarrow \text{số nút} = 0$  (Bù)

$\text{num} = 1025 \rightarrow \text{số nút} = 8$

### 2.5. Bài 5 (Data encryption and Decryption)

- **Mô tả:**



Một công ty muốn truyền dữ liệu qua đường mạng tuy nhiên họ sợ rằng dữ liệu sẽ bị đánh cắp. Công ty thuê một lập trình viên để giải quyết vấn đề này. Sau khi xem xét lập trình viên quyết định lựa chọn chiến lược sẽ mã hóa (encrypt) dữ liệu trước khi truyền trên mạng bởi vì nếu dữ liệu đã được mã hóa thì nếu có bị đánh cắp thì cũng không biết nó có ý nghĩa gì. Dữ liệu được gói đi được chia thành nhóm gồm 4 chữ số.

Ứng dụng của lập trình viên sẽ đọc một số nguyên có 4 chữ số. Sau đó số này được mã hóa theo các bước như sau: (1) thay thế mỗi kí số bởi tổng của nó với 7 và lấy phần dư cho 10 (2) Đảo vị trí của kí số thứ 1 với kí số thứ 3, kí số thứ 2 và kí số thứ 4.

Ví dụ: số ban đầu là 1093 sẽ được mã hóa thành 6087. Kết quả mã hóa sau đó được in ra màn hình.

**Yêu cầu:** Hãy viết hai chương trình

- Chương trình 1: mã hóa một số được nhập vào chương trình
- Chương trình 2: giả mã (decryption) một số nhập vào để tính ra giá trị ban đầu.

## 2.6. Bài 6:

Lập trình cho phép nhập vào từ bàn phím 2 số nguyên dương a và b. Sau đó xuất ra màn hình số lớn nhất.

## 2.7. Bài 7:

Viết chương trình cho người dùng nhập vào ba số khác không. Sau đó kiểm tra xem chúng có tạo thành ba cạnh của tam giác hay không?

## 2.8. Bài 8:

Viết chương trình cho người dùng nhập vào ba số nguyên dương. Sau đó kiểm tra xem chúng có tạo thành ba cạnh của một tam giác hay không, nếu có xuất ra loại của tam giác: thường, vuông, cân, đều ?

## 2.9. Bài 9:

Xác định lỗi trong đoạn chương trình sau:

(Chú ý: Có thể có nhiều hơn một lỗi trong mỗi đoạn chương trình)

a)

```
1. if ( tuoi >= 65 );
```



```
2. System.out.println( "Tuoi >= 65" );
3. else
4. System.out.println( "Tuoi < 65 )" );
```

b)

```
1. switch ( songuyen % 2 ){
2.     case 0:
3.         System.out.println( "So nguyen chan" );
4.
5.     case 1:
6.         System.out.println( "So nguyen le" );
7. }
```

## 2.10. Bài 10: (Dangling-else Problem)

Xác định kết quả của các đoạn mã bên dưới với trường hợp:

(I)  $x = 9$  và  $y = 11$

(II)  $x = 11$  và  $y = 9$ .

Sau khi hoàn tất bài toán hãy rút ra bài học cho việc viết chương trình sử dụng Java.

a)

```
1. if ( x < 10 )
2. if ( y > 10 )
3. System.out.println( "*****" );
4. else
5. System.out.println( "#####" );
6. System.out.println( "$$$$$" );
```

b)

```
1. if ( x < 10 )
2. {
3.     if ( y > 10 )
4.     System.out.println( "*****" );
5. }
```



6. `else`
7. `{`
8. `System.out.println( "####" );`
9. `System.out.println( "$$$$" );`
10. `}`

### 2.11. Bài 11:

Để xác định nhiệt độ của môi trường, thông thường người ta thường sử dụng thang độ Celsius ( $^{\circ}\text{C}$ ). Tuy nhiên, trong khoa học, các nhà khoa học sử dụng hệ Fahrenheit ( $^{\circ}\text{F}$ ). Công thức để chuyển từ thang độ Celsius sang Fahrenheit:

$$C = 5.0 / 9.0 * ( F - 32 );$$

Ngược lại, để chuyển từ độ Fahrenheit sang hệ Celsius ta sử dụng công thức:

$$F = 9.0 / 5.0 * C + 32;$$

Hãy viết chương trình cho phép người dùng nhập vào thang độ hiện tại cùng với giá trị nhiệt độ hiện tại. Sau đó tính toán và in ra nhiệt độ trong thang độ còn lại.

### 2.12. Bài 12:

Giải bài toán tìm nghiệm phương trình bậc 2:  $ax^2 + bx + c = 0$ , với  $a \neq 0$

### 2.13. Bài 13:

Lập trình cho phép nhập vào từ bàn phím điểm các môn học Toán, Lý, Hoá. Tính điểm trung bình và xuất ra màn hình thông tin theo yêu cầu sau:

Nếu điểm trung bình  $\geq 5 \rightarrow$  "đạt", ngược lại: "không đạt"

Biết rằng: Toán có hệ số 3, Lý có hệ số 2, Hoá có hệ số 1

### 2.14. Bài 14:

Lập trình cho phép nhập vào từ bàn phím tháng, năm. Xét tháng năm vừa nhập có hợp lệ hay không. Nếu hợp lệ  $\rightarrow$  xuất ra màn hình số ngày của tháng.

Ghi chú:

- Nếu năm nhuận thì tháng 2 có 29 ngày, ngược lại có 28 ngày. Tháng 1, 3, 5, 7, 8, 10, 12: có 31 ngày. Ngoài ra, các tháng còn lại có 30 ngày.
- Năm nhuận là năm chia hết cho 4 và không chia hết cho 100 hoặc chia hết cho 400.



### 2.15. Bài 15:

Lập trình cho phép nhập vào từ bàn phím các thông tin sau:

1. Phép toán cần thực hiện (1. cộng ; 2. trừ ; 3. nhân; 4. chia)
2. Các số hạng : so\_hang1, so\_hang\_2 là các số thực

**Chú ý:** trong trường hợp phép toán đã chọn là phép chia thì so\_hang\_2  $\neq 0$

**Yêu cầu:** tính toán và xuất ra màn hình kết quả tương ứng

**Mở rộng:** cho phép người dùng lặp lại xử lý

### 2.16. Bài 16:

Tìm và sửa các lỗi trong mỗi đoạn chương trình sau:

a)

1. `for ( i = 100, i >= 1, i++ )`
2. `System.out.println( i );`

b)

1. `for ( i = 19; i >= 1; i += 2 )`
2. `System.out.println( i );`

c)

1. `counter = 2;`
2. `do{`
3. `System.out.println( counter );`
4. `counter += 2 ;`
5. `} While ( counter < 100 );`

d)

1. `int x = 1, total;`
2. `while ( x <= 10 ){`
3. `total += x;`
4. `++x;`
5. `}`

e)



1. `while ( x <= 100 )`
2. `total += x;`
3. `++x;`

f)

1. `while ( y > 0 )`
2. `{`
3. `System.out.println( y );`
4. `++y;`

### 2.17. Bài 17:

Viết chương trình Java sử dụng vòng lặp, cho phép người dùng nhập số N và in ra bảng có dạng sau:

N	10*N	100*N	1000*N
1	10	100	1000
2	20	200	2000
3	30	300	3000
4	40	400	4000
5	50	500	5000

Hình 1: Kết quả N=5

### 2.18. Bài 18:

Viết chương trình cho phép nhập một số nguyên dương chỉ bao gồm 0 và 1 (ví dụ: số nhị phân) sau đó tính và in giá trị trong hệ thập phân tương ứng

Gợi ý: Sử dụng toán tử chia và phần dư để lần lượt lấy từng ký số từ phải sang trái. Sau đó nhân từng ký số này với trong số tương ứng với vị trí của số đó. Ví dụ ta có số nhị phân: 1101 thì giá trị thập phân tương ứng là:  $1 * 1 + 0 * 2 + 1 * 4 + 1 * 8 = 13$

Phát triển bài toán trên cho phép một số cùng hệ số của nó sau đó in giá trị thập phân của nó.

### 2.19. Bài 19:





Lập trình cho phép nhập vào từ bàn phím số  $n > 0$  bất kỳ. Tính số nút của  $n$ .

## 2.20. Bài 20:

Lập trình cho phép nhập vào từ bàn phím số nguyên dương  $n$ .

Xét  $n$  có phải là số nguyên tố.

Phát triển bài toán trên để in ra tất cả các số nguyên tố nhỏ hơn  $n$

Ghi chú: số nguyên tố là số nguyên dương chỉ chia hết cho 1 và chính nó

## 2.21. Bài 21:

Trong Java, để sinh ra một số ngẫu nhiên (nghĩa là giá trị của nó không biết trước khi chạy giống như việc gieo con súc sắc) người ta sử dụng phương thức `random()` của lớp `Math`. Phương thức `random()` sẽ sinh ra một số thực ngẫu nhiên có giá trị trong khoảng **[0,1)**. Ví dụ sau cho biết cách một số nguyên ngẫu nhiên trong đoạn  $[0,6)$ :

```
int face = (int)(Math.random() * 6)
```

Sử dụng các kiến thức vừa trình bày ở trên để tính xác suất xuất hiện có của mỗi mặt của con súc sắc có 6 mặt nếu gieo con súc sắc 1000 lần sau đó xuất kết quả này ra màn hình

Ví dụ kết quả kết xuất:

1: 25%

2: 15%

3: 8%

4: 12.5%

5: 27%

6: 12.5%

## 2.22. Bài 21:

Viết chương trình nhập vào số  $N$  và in ra các hình sau (ví dụ  $n = 4$ )



a)

```

*
**
***
****

```

a')

```

****
***
**
*

```

b)

```

  *
 **
***
****

```

b')

```

****
***
**
*

```

c)

```

  *
 ***
****
*****

```

c')

```

*****
****
***
**

```



## BÀI 3: Array và String



Giúp học viên làm quen với các thao tác xử lý cơ bản trên mảng và các thao tác xử lý chuỗi

### 3.1. Bài 1

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Xuất ra màn hình mảng a vừa nhập.

### 3.2. Bài 2

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Đếm số lượng phần tử chẵn/ lẻ của mảng

### 3.3. Bài 3

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Nhập 1 số nguyên x, đếm số lần xuất hiện của x trong mảng.

### 3.4. Bài 4

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Tính trung bình cộng các phần tử chẵn/ lẻ của mảng.

### 3.5. Bài 5

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Đếm số phần tử là số nguyên tố trong mảng.

### 3.6. Bài 6

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử dương. Nhập vào 1 số nguyên x. Nhập vào vị trí cần chèn x vào mảng a. Chèn x vào mảng a tại vị trí chỉ định, xuất mảng a ra màn hình.

### 3.7. Bài 7



Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Sắp xếp mảng a theo thứ tự tăng dần/ giảm dần

### 3.8. **Bài 8**

Viết chương trình cho phép nhập từ bàn phím mảng a có n phần tử nguyên. Xuất ra màn hình phần tử dương nhỏ nhất và phần tử âm lớn nhất.

Lưu ý: xuất thông báo nếu không tồn tại phần tử thỏa yêu cầu.

### 3.9. **Bài 9:**

Nhập vào một chuỗi các số cách nhau bởi khoảng trắng, ví dụ: 1 21 3 5 7. Xử lý chuỗi nhập để tạo ra mảng các số

Yêu cầu:

- Sử dụng các phương thức của lớp String
- Sử dụng StringTokenizer

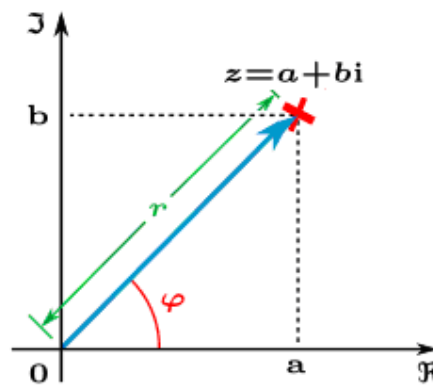
## BÀI 4: Lập trình hướng đối tượng với Java



Giúp học viên biết cách khai báo class, property, method. Cách khởi tạo object.  
Áp dụng các toán tử để thực hiện các phép toán

### 4.1. Bài 1:

Trong toán học, một số phức là một số được tạo thành từ hai thành phần: phần thực (real) và phần ảo (imaginary) và được viết dưới dạng  $a + bi$ , trong đó  $a$  và  $b$  là những số thực, và  $i$  là một số đặc biệt thỏa điều kiện  $i^2 = -1$ .



Hình 1: Biểu diễn số phức trong mặt phẳng

Một số phức  $x = a + bi$  có thể biểu diễn trên mặt phẳng tọa độ như trong Hình 1. Trục hoành được gọi là trục thực và trục tung được gọi là trục ảo. Với cách biểu diễn của số phức trên mặt phẳng, gọi khoảng cách (modulus) từ số phức đến gốc tọa độ là  $r$  để dàng nhận thấy  $r = \sqrt{a^2 + b^2}$ . Góc (argument) kí hiệu  $\varphi$  là góc hợp bởi số phức so với trục hoành.

Có thể dễ dàng áp dụng các phép toán cơ bản lên số phức. Các số phức có thể được cộng, trừ, nhân và chia với nhau để tạo nên một số phức khác

Giả sử có hai số phức  $x = a + bi$  và  $y = c + di$ . Tổng hai số phức này là một số phức có dạng:

$$\begin{aligned} z &= (a + bi) + (c + di) \\ &\Rightarrow (a + c) + (b + d)i \end{aligned} \quad (1)$$

Tương tự, phép trừ hai số phức, cho bởi công thức



$$\begin{aligned} z &= (a + bi) - (c + di) \\ \Rightarrow (a - c) + (b - d)i \end{aligned} \quad (2)$$

Để tiến hành phép nhân hai số phức, đầu tiên chúng ta cần khai triển tích hai số phức đồng thời chú ý  $i^2 = -1$ . Sau đó tiến hành đặt thừa số chung, kết quả được cho trong (3)

$$\begin{aligned} z &= (a + bi) * (c + di) \\ \Rightarrow ac + adi + bci + bdi^2 \\ \Rightarrow (ac - bd) + (ad + bc)i \end{aligned} \quad (3)$$

Để thực hiện phép chia hai số phức, chú ý đẳng thức  $a^2 - b^2 = (a - b)(a + b)$ . Dựa vào nhận xét trên để tiến hành chia hai số phức, ta làm như sau:

$$\begin{aligned} z &= \frac{a + bi}{c + di} \\ \Rightarrow \frac{(a + bi)(c - di)}{(c + di)(c - di)} \\ \Rightarrow \frac{ac + bd}{c^2 + d^2} + \frac{bc - ad}{c^2 + d^2}i \end{aligned} \quad (4)$$

Với những mô tả về số phức như trên, học viên sử dụng ngôn ngữ Java, hãy khai báo lớp Complex với các đặc điểm (property) và phương thức (method) để thực hiện các phép toán trên số phức.

Để kiểm tra các phương thức đã được viết, học viên tạo ra 2 số phức để thử nghiệm các phương thức đã hiện thực

#### 4.2. Bài 2:

Tạo một lớp **Nhanvien** gồm ba thông tin là: **honv** (kiểu String), **tennv** (kiểu String) và **luong** (kiểu double). Lớp này có một constructor (hàm tạo) để khởi tạo ba thông tin này. Cung cấp phương thức set và get (set and get method) cho mỗi thông tin này. Nếu **luong** nhỏ hơn 0, thì gán thành 0.0. Viết chương trình để tạo hai đối tượng **Nhanvien** và hiển thị các thông tin của đối tượng này. Sau đó tăng lương của mỗi nhân viên 10% và lặp lại quá trình in thông tin nhân viên.

#### 4.3. Bài 3



Tạo một class tên là **Date** có các thông tin là: **month** (kiểu int), **day** (kiểu int) và **year** (kiểu int). Lớp này có một constructor (hàm tạo) để khởi tạo các thông tin này. Cung cấp phương thức set và get cho các thông tin này. Ngoài ra, lớp này có thêm phương thức **displayDate** để hiển thị month, day và year theo định dạng month/day/year (ví dụ: 9/11/2009).

Viết chương trình để tạo ra một đối tượng của lớp Date vừa hiển thực sau đó gọi các phương thức vừa hiển thực

#### 4.4. **Bài 4**

Xây dựng lớp Diem2D trong mặt phẳng tọa độ Oxy

- Thuộc tính: x, y là tung độ và hoành độ
- Phương thức:
  - Nhap(): nhập tung độ và hoành độ
  - Xuat(): xuất tung độ và hoành độ
  - Xuất theo định dạng (x, y) (VD: khi x = 5 và y = 8 thì xuất ra (5,8))
  - TinhKhoangCach(): tính khoảng cách từ điểm hiện hành tới điểm khác

Tính căn bậc hai: dùng phương thức sqrt() của lớp Math (ví dụ: Math.sqrt(8) sẽ là ~2.83)

Công thức tính khoảng cách giữa 2 điểm  $A(x_a, y_a)$ ,  $B(x_b, y_b)$  trong mặt phẳng tọa độ Oxy

Khoảng cách =  $\text{Math.sqrt}((x_a - x_b) * (x_a - x_b) + (y_a - y_b) * (y_a - y_b))$

#### 4.5. **Bài 5**

Xây dựng lớp HìnhTron2D trong mặt phẳng tọa độ Oxy

- Thuộc tính: O là điểm tâm, R là điểm trên hình tròn có kiểu Diem2D (bán kính là khoảng cách từ O đến R)
- Phương thức:
  - Nhap(): nhập các điểm O và R
  - Xuat(): xuất các đỉnh
  - TinhChuVi(): tính chu vi hình tròn
  - TinhDienTich(): tính diện tích hình tròn

#### 4.6. **Bài 6**



Xây dựng lớp TamGiac2D trong mặt phẳng tọa độ Oxy

- Thuộc tính: A, B, C là các đỉnh có kiểu Diem2D
- Phương thức:
  - Nhap(): nhập các đỉnh
  - Xuat(): xuất các đỉnh
  - TinhChuVi(): tính chu vi tam giác
  - TinhDienTich(): tính diện tích tam giác

#### 4.7. **Bài 7**

Xây dựng lớp HinhChuNhat2D trong mặt phẳng tọa độ Oxy

- Thuộc tính: A, C là hai đỉnh đối diện nhau có kiểu Diem2D (2 cạnh của hình chữ nhật song song với 2 trục toạ độ)
- Phương thức:
  - Nhap(): nhập các đỉnh
  - Xuat(): xuất các đỉnh
  - TinhChuVi(): tính chu vi hình chữ nhật
  - TinhDienTich(): tính diện tích hình chữ nhật





# BÀI 5: Tính thừa kế (Inheritance) và tính đa hình (Polymorphism)



Giúp học viên thành thạo trong việc xây dựng các class kế thừa từ class khác cũng như hiểu rõ tính đa hình của class.

## 5.1. Bài 1:

Hiện thực mối quan hệ kế thừa giữa Hình2D, HìnhTronD, HìnhTamGiac2D, HìnhChuNhat2D và HìnhVuong2D. Cung cấp các thuộc tính, các phương thức in thông tin, tính diện tích, chu vi và áp dụng tính đa hình cho bài toán trên.

## 5.2. Bài 2:

Xây dựng ứng dụng quản lý lương của công ty Z. Thông tin chung của nhân viên gồm: mã NV, họ NV, tên NV, ngày sinh, địa chỉ, ngày vào làm.

Lương được tính như sau:

Nhân viên văn phòng: lương cơ bản x hệ số lương

Nhân viên sản xuất: số sản phẩm x đơn giá

Với các nhân viên có con < 3 tuổi: trợ cấp 5% lương.

**Yêu cầu:** nhập danh sách nhân viên (gồm cả nhân viên văn phòng và nhân viên sản xuất), tính lương và xuất ra kết quả như sau:

Nhân viên: 001, Họ tên: Nguyễn Văn A , Ngày sinh: 1/1/1975, Ngày vào làm: 2/2/2000

Loại NV: NVVP, Lương: 3500000

Nhân viên: 002, Họ tên: Nguyễn Văn B , Ngày sinh: 12/5/1980, Ngày vào làm: 6/28/2005

Loại NV: NVSX, Lương: 2500000

Nhân viên: 003, Họ tên: Nguyễn Văn C , Ngày sinh: 6/14/1982, Ngày vào làm: 4/16/2004

Loại NV: NVVP, Lương: 2500000



## BÀI 6: Exception



*Giúp học viên biết cách xử lý lỗi khi chương trình có khả năng xảy ra lỗi*

### 6.1. Bài 1:

Thêm mã xử lý lỗi cho các đoạn code có khả năng xảy ra lỗi, áp dụng vào các bài tập trên