

## Chuyên đề 2

### Lệnh điều kiện và vòng lặp

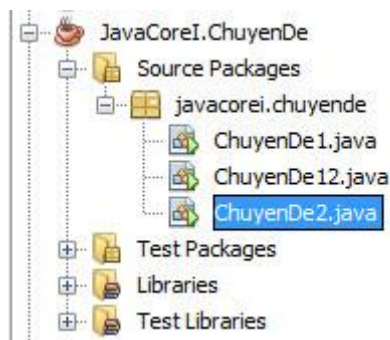
#### Mục tiêu

- ✓ Sử dụng được 2 loại mệnh đề điều kiện
- ✓ Sử dụng được 3 loại vòng lặp.

#### Bài thực hành số 1:

**Yêu cầu:** tạo project Java và file ChuyenDe2.java có hàm main như hình dưới. Viết mã nguồn thực hiện các yêu cầu như sau:

- Tìm UCLN và BCNN của hai số a và b cho trước
- In ra chuỗi Fibonacci của một số n cho trước
- Liệt kê tất cả các số nguyên tố nhỏ hơn số n ở trên
- Có một số, cho biết đó có phải là số Thuận/Nghịch không. VD: 123321 đọc xuôi/ngược đều như nhau => là số Thuận/Nghịch.



```
Output - JavaCoreI.ChuyenDe (run)

run:
UCLN = 5
BCNN = 200

Dãy Fibonacci
0 1 1 2 3 5 8 13
Số Nguyên Tố:
2 3 5 7 11 13 17 19
Số Thuận/Nghịch
soKT1 là số THUẬN/NGHỊCH
soKT2 KHÔNG PHẢI là số THUẬN/NGHỊCH
BUILD SUCCESSFUL (total time: 0 seconds)
```

### Code tham khảo:

`package javacorei.chuyende;`

```
/**
 *
 * @author minhvufo
 */
public class ChuyenDe2 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // Tìm UCLN
        int a = 25, b = 40, ucln = 0;
        if (a == 0 && b == 0) {
            // Nếu cả a và b = 0
            System.out.println("Không có UCLN vì a và b = 0");
        } else if (a == 0 || b == 0) {
            // Nếu a hoặc b = 0 thì UCLN bằng trị tuyệt đối số # 0
            System.out.println("UCLN = " + Math.abs(a - b));
        } else {
```

// Giải thuật Euclid: chạy vòng lặp lấy số LỚN hơn trừ BÉ hơn cho đến khi BẰNG nhau

```
while (a != b) {
    if (a > b) {
        a = a - b;
    } else {
        b = b - a;
    }
}
ucln = a; // b
System.out.println("UCLN = " + ucln);
}
```

// Tìm BCNN

```
a = 25;
b = 40;
if (a == 0 || b == 0) {
    System.out.println("BCNN = 0");
} else {
    // Công thức tính BCNN: BCNN = |a.b| / UCLN
    int bcnn = Math.abs(a * b) / ucln;
    System.out.println("BCNN = " + bcnn);
}
```

// Dãy Fibonacci

```
int n = 20, x = 0, y = 1; // Bắt đầu chuỗi Fibonacci là 0-1 hoặc 1-1
System.out.println("\nDãy Fibonacci");
System.out.print(x + " " + y + " "); // In trước 2 số đầu tiên
for (int i = x + y; i <= n; i = x + y) {
    System.out.print(i + " ");
    // Thay đổi giá trị
    x = y;
    y = i;
}
```

// Liệt kê số Nguyên Tố: là số chia hết cho 1 và CHÍNH nó

// Số 1 không phải là số nguyên tố, số 2 là số Nguyên tố nhỏ nhất => vòng lặp từ 2

```
System.out.println("\nSố Nguyên Tố: ");
for (int i = 2; i <= n; i++) {
    boolean snt = true;
    for (int j = 2; j < i; j++) {
        if (i % j == 0) { // Phát hiện chia hết cho số khác trong khoảng 1->i
            snt = false;
        }
    }
}
```

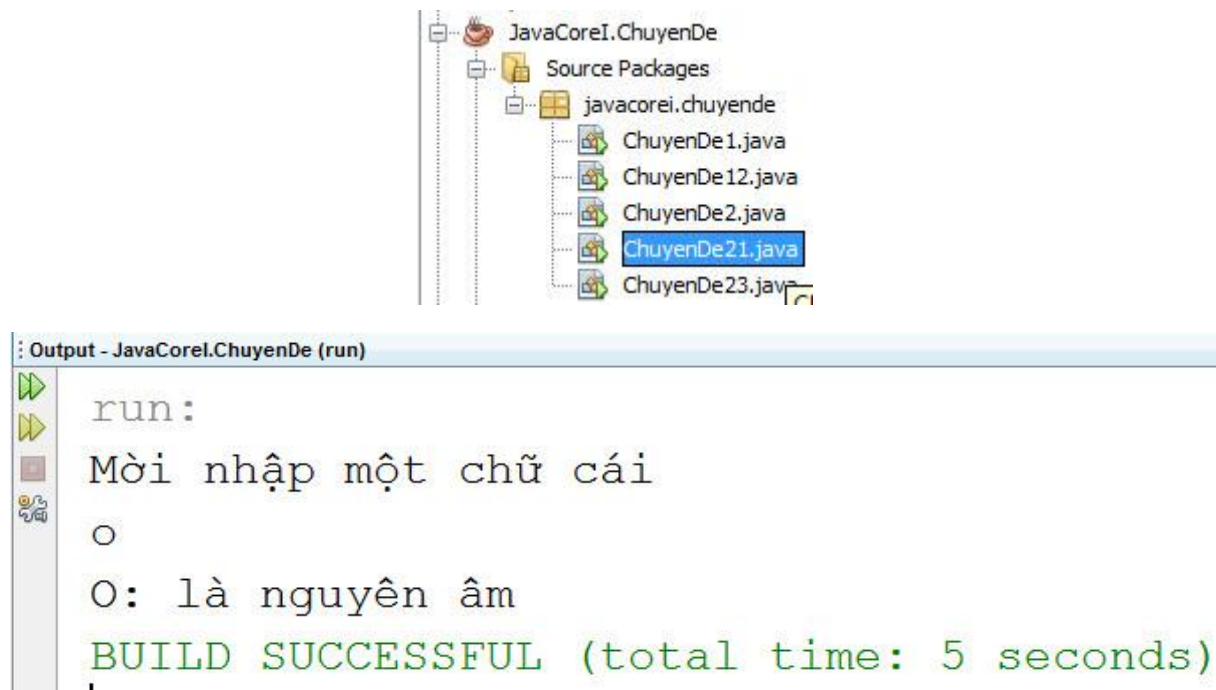
```
        break;
    }
}
if (snt) {
    System.out.print(i + " ");
}
}

// Kiểm tra số thuận/ngịch
System.out.println("\nSố Thuận/Nghịch");
int soKT1 = 123454321, soKT2 = 1234500;
StringBuilder sb = new StringBuilder();
sb.append(String.valueOf(soKT1)); // Chuyển đổi soKT1 về chuỗi rồi gán vào
StringBuilder
String soKT1_dao = sb.reverse().toString();
if (String.valueOf(soKT1).equals(soKT1_dao)) {
    System.out.println("soKT1 là số THUẬN/NGHỊCH");
} else {
    System.out.println("soKT1 KHÔNG PHẢI là số THUẬN/NGHỊCH");
}

// Làm tương tự để kiểm tra soKT2
sb.delete(0, sb.length()); // Reset StringBuilder
sb.append(String.valueOf(soKT2)); // Chuyển đổi soKT1 về chuỗi rồi gán vào
StringBuilder
String soKT2_dao = sb.reverse().toString();
if (String.valueOf(soKT2).equals(soKT2_dao)) {
    System.out.println("soKT2 là số THUẬN/NGHỊCH");
} else {
    System.out.println("soKT2 KHÔNG PHẢI là số THUẬN/NGHỊCH");
}
}
}
```

## Bài thực hành số 2:

**Yêu cầu:** viết chương trình yêu cầu nhập vào một chữ cái, cho biết đó có phải là nguyên âm hay không.



### Code tham khảo:

```
package javacorei.chuyende;
```

```
import java.util.Scanner;
```

```
/**
```

```
 *
```

```
 * @author minhvuvc
```

```
 */
```

```
public class ChuyenDe21 {
```

```
    /**
```

```
     * @param args the command line arguments
```

```
     */
```

```
    public static void main(String[] args) {
```

```
        // Nhận biết nguyên âm: a, e, i, o, u
```

```
        Scanner sc = new Scanner(System.in);
```

```
        System.out.println("Mời nhập một chữ cái");
```

```
        char nguyenAm = sc.nextLine().charAt(0);
```

```
        switch (nguyenAm) {
```

```
            case 'a':
```

```
            case 'e':
```

```
            case 'i':
```

```
            case 'o':
```

```
            case 'u':
```

```

        System.out.println(String.valueOf(nguyenAm).toUpperCase() + ": là
nguyên âm");
        break;
    default:
        System.out.println(nguyenAm + ": KHÔNG là nguyên âm");
        break;
    }
}
}

```

### Bài thực hành số 3:

**Yêu cầu:** Viết chương trình Java core vẽ hình đồng hồ cát như sau:



**Lưu ý:** số lượng \* của cạnh tam giác cân có thể tùy ý (như hình là 11).

### Code tham khảo:

```
package javaapplication1;
```

```

/**
 *
 * @author minhvufo
 */

```

```
public class DongHoCat {
```

```

/**
 * @param args the command line arguments
 */
public static void main(String[] args) {
    int size = 10;      // Kích thước cạnh tam giác
    int pivot = 10 / 2; // Tâm điểm
    int cnt = pivot - 0; // Biến đếm vị trí. VD: 5 - 0 = 5
    for (int i = 0; i <= size; i++) {    // DÒNG
        for (int j = 0; j <= size; j++) { // CỘT
            // Khi cnt dần giảm giá trị => tam giác xuôi
            if (cnt >= 0) {
                if (j >= pivot - cnt && j <= pivot + cnt) {
                    System.out.print("*");
                } else {
                    System.out.print(" ");
                }
            }
            /*
             Phải nằm trong phạm vi quy định mới in ra *
            *****
            1***6
            12*56
            */

        }
        // Khi cnt có giá trị âm => chuyển tam giác ngược
        if (cnt < 0) {
            int newCNT = -cnt; // Đổi về giá trị dương cho dễ hiểu
            if (j >= pivot - newCNT && j <= pivot + newCNT) {
                System.out.print("*");
            } else {
                System.out.print(" ");
            }
        }
        /*
             Phải nằm trong phạm vi quy định mới in ra *
            12*56
            1***6
            *****
            */

    }
}
System.out.println("");
cnt--; // Giảm dần sau mỗi dòng 5 -> 4 -> 3....
}
}
}

```