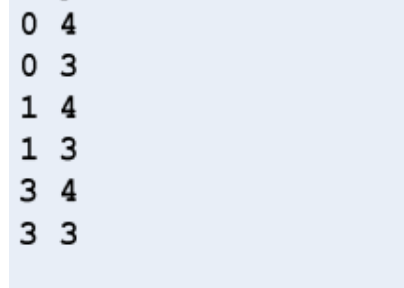


LẬP TRÌNH CƠ BẢN VỚI JAVA

TUẦN 3. CẤU TRÚC ĐIỀU KHIỂN

1. Sắp xếp lại các dòng mã sau đây thành chương trình có kết quả hiển thị như hình dưới. Tự bổ sung các ngoặc đóng } vào những nơi thích hợp

```
x++;  
if (x==1){  
for(int x = 0; x < 4; x++){  
public class MultiFor{  
for(int y = 4; y > 2; y --){  
System.out.println(x + " " + y);  
Public static void main(String args[ ]){
```



```
0 4  
0 3  
1 4  
1 3  
3 4  
3 3
```

2. Viết các lệnh Java (không cần viết đủ chương trình) để thực hiện từng nhiệm vụ sau đây:
 - a. Tính tổng các số chẵn trong khoảng từ 1 đến 99.
 - b. Sử dụng vòng lặp để in ra các số từ 1 đến 10 trên một dòng, dùng kí tự tab ('\t') để ngăn cách giữa các số.
3. Viết chương trình thử nghiệm việc đọc và in dữ liệu (ra màn hình): Tên (`String`), tuổi (`int`), lương (`float`). Sử dụng vòng lặp để thực hiện điều này 10 lần.

Trong các bài tập dưới đây, nếu có yêu cầu viết chương trình, bạn hãy thực hiện việc viết dưới dạng chương trình con. Hàm main chỉ phục vụ thông báo, nhập dữ liệu và gọi các chương trình con.

4. Xây dựng hàm `isPrime(int K)` kiểm tra xem một số `K` kiểu `int` có là nguyên tố hay không, sau đó viết chương trình cho phép nhập số nguyên `N`, rồi liệt kê `N` số nguyên tố đầu tiên ra màn hình.
5. Số hoàn thiện (hoàn hảo) là số tự nhiên bằng tổng các ước số dương chính thức (tức không tính bản thân số đó) của nó. Ví dụ: $6 = 1 + 2 + 3$; $28 = 1 + 2 + 4 + 7 + 14$.
 - a. Hãy xây dựng hàm `isPerfectNumber` kiểm tra xem một đối số `K` kiểu `int` có là số hoàn thiện hay không.
 - b. Viết chương trình cho phép nhập số nguyên dương `N` từ bàn phím, sau đó liệt kê `N` số hoàn thiện đầu tiên (nhỏ nhất) ra màn hình.
6. Một số nguyên dương được gọi là đối xứng nếu viết các chữ số của nó theo chiều ngược lại vẫn được chính số đó. Ví dụ: 1, 2, ..., 11, 22, ... 101, 121, ... là số đối xứng.

- a. Viết hàm isPalindrome, với đối số K kiểu int, kiểm tra xem K có phải là số đối xứng hay không, trả về đúng hoặc sai.
 - b. Viết chương trình cho phép nhập hai số nguyên dương M và N, sau đó liệt kê các số đối xứng ở giữa các số M và N.
7. Một số nguyên dương có k chữ số được gọi là số Armstrong nó đúng bằng tổng lũy thừa k của các chữ số của nó. Ví dụ: 371 là số Armstrong có 3 chữ số vì $3^3 + 7^3 + 1^3 = 371$; 1634 là số Armstrong có 4 chữ số vì $1^4 + 6^4 + 3^4 + 4^4 = 1634$.
 - a. Viết hàm isArmstrong, với đối số nguyên K, kiểm tra xem K có là số Armstrong hay không.
 - b. Viết chương trình cho phép nhập một số nguyên dương N, sau đó in ra N số Armstrong đầu tiên.
8. Xây dựng lớp (class) NumberProperties, trong lớp này có các hàm sau:
 - a. IsPrime: Kiểm tra một số nguyên dương N có phải là nguyên tố hay không.
 - b. IsPalindrome: Kiểm tra một số nguyên dương N có phải là số đối xứng hay không.
 - c. IsArmstrong: Kiểm tra một số nguyên dương N có phải là số Armstrong hay không.
 - d. IsPerfectNumber: Kiểm tra một số nguyên dương có là số hoàn thiện hay không.
 - e. Hàm main: Nhận một số nguyên dương N qua đối dòng lệnh, sau đó in ra màn hình:
 - i. N số nguyên tố đầu tiên
 - ii. N số đối xứng đầu tiên
 - iii. N số Armstrong đầu tiên
 - iv. N số hoàn thiện đầu tiên.

9. Cho số thực (double) $x > 0$. Công thức xấp xỉ e^x theo khai triển Taylor như sau

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$$

Viết chương trình (class) Exp.java, trong đó có một hàm dblExp tính và trả lại giá trị e^x theo công thức trên, với x (double) và n (int) là 2 đối số. Trong hàm main, cho phép nhập x, n từ bàn phím, gọi dblExp và in ra giá trị gần đúng e^x .

10. Xây dựng chương trình (class) TrigonometricFunctions, có các hàm sau

- a. Hàm tính gần đúng $\sin x$ theo khai triển Taylor:

$$\sin(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} \dots + \frac{(-1)^{(n+1)} x^{(2n-1)}}{(2n-1)!}$$

b. Hàm tính gần đúng cosin x theo khai triển Taylor

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} \dots + \frac{(-1)^n x^{2n}}{(2n)!}$$

Trong hai hàm trên, các đối số là x và $\varepsilon > 0$ (sai số - cùng kiểu double).

c. Hàm main: Cho phép nhập giá trị đối số x và sai số $\varepsilon > 0$. Sử dụng các hàm trên tính và in ra giá trị gần đúng của sin x , cosin x với sai số ε .

11. Giả sử m, d, y (kiểu int) là giá trị tháng (tính theo tháng 1, 2, 3, ..., 12), ngày (1, ..., 31) và năm của một ngày nào đó. Ta có thể xác định đó là ngày thứ mấy trong tuần theo công thức sau (ký hiệu $/$ là phép chia nguyên – lấy thương)

$$y_0 = y - (14 - m)/12;$$

$$x = y_0 + y_0/4 - y_0/100 + y_0/400;$$

$$m_0 = m + 12 \times ((14 - m)/12) - 2;$$

$$d_0 = (d + x + 31m_0/12) \bmod 7;$$

sau đó, giá trị của d_0 tương ứng với ngày trong tuần như sau: 0 ứng với thứ hai, 1 ứng với thứ ba, ..., 6 ứng với chủ nhật.

Xây dựng một lớp, trong đó có một hàm DayofWeek, tham số đầu vào là các đối nguyên (int) m, d, y . Hàm trả lại xâu ký tự ứng với thứ trong tuần của ngày $d/m/y$. Hàm main cho phép nhập các biến nguyên d, m, y từ bàn phím, sau đó gọi hàm DayofWeek và in ra màn hình đó là thứ mấy trong tuần.