

Chương 3: Cấu trúc điều khiển

Kiểu boolean và Các toán tử

Kết quả của phép so sánh là 1 giá trị **Boolean**:
true hoặc **false**.

```
boolean b = (1 > 2);
```



Toán tử quan hệ

Toán tử Java	Ký hiệu Toán học	Mô tả	Ví dụ (radius = 5)	Kết quả
<	<	nhỏ hơn	<code>radius < 0</code>	<code>false</code>
<=	\leq	nhỏ hơn hoặc bằng	<code>radius <= 0</code>	<code>false</code>
>	>	lớn hơn.	<code>radius > 0</code>	<code>true</code>
\geq	\geq	lớn hơn hoặc bằng	<code>radius >= 0</code>	<code>true</code>
$=$	$=$	bằng	<code>radius == 0</code>	<code>false</code>
\neq	\neq	khác	<code>radius != 0</code>	<code>true</code>



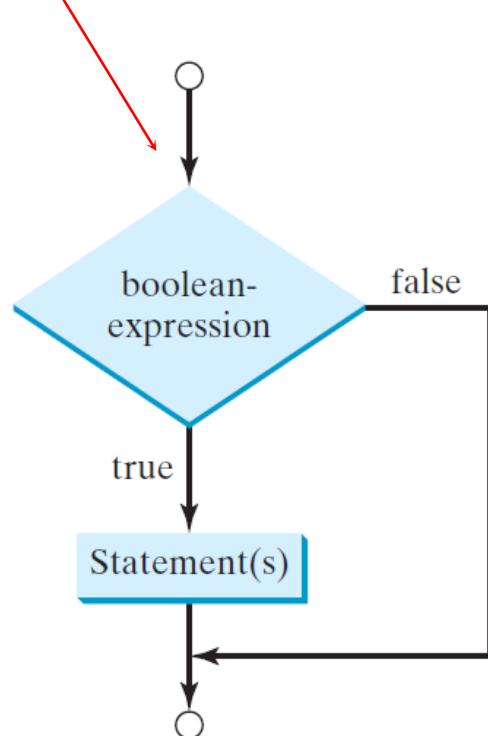
Bài toán: Công cụ học toán đơn giản

Chương trình phát sinh ngẫu nhiên 2 số nguyên number1 và number2, và hiển thị lên màn hình câu hỏi chẳng hạn như “Kết quả của $7 + 9$ là bao nhiêu?”. Sau đó học sinh gõ câu trả lời, chương trình sẽ hiển thị một thông báo cho biết câu trả lời đó là đúng hay sai.

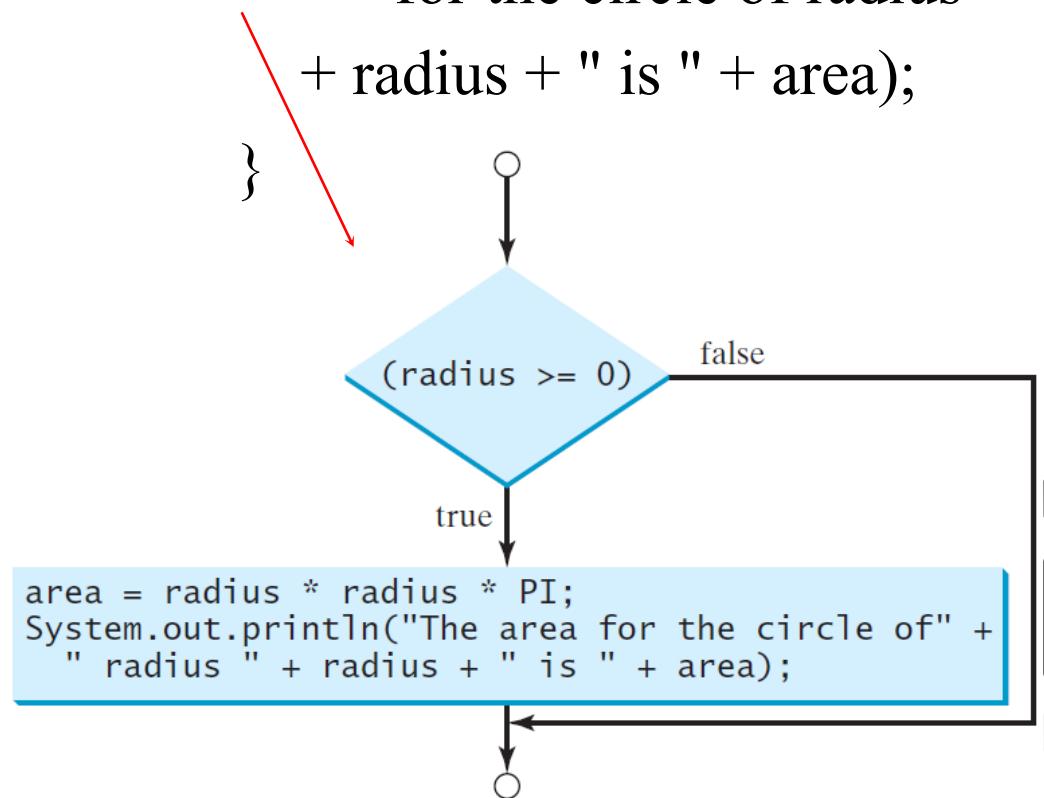


Câu lệnh if

```
if (boolean-expression) {  
    statement(s);  
}
```



```
if (radius >= 0) {  
    area = radius * radius * PI;  
    System.out.println("The area"  
        + " for the circle of radius "  
        + radius + " is " + area);  
}
```



Lưu ý

```
if i > 0 {  
    System.out.println("i is positive");  
}
```

(a) Wrong

```
if (i > 0) {  
    System.out.println("i is positive");  
}
```

(b) Correct

```
if (i > 0) {  
    System.out.println("i is positive");  
}
```

(a)

Equivalent

```
if (i > 0)  
    System.out.println("i is positive");
```

(b)



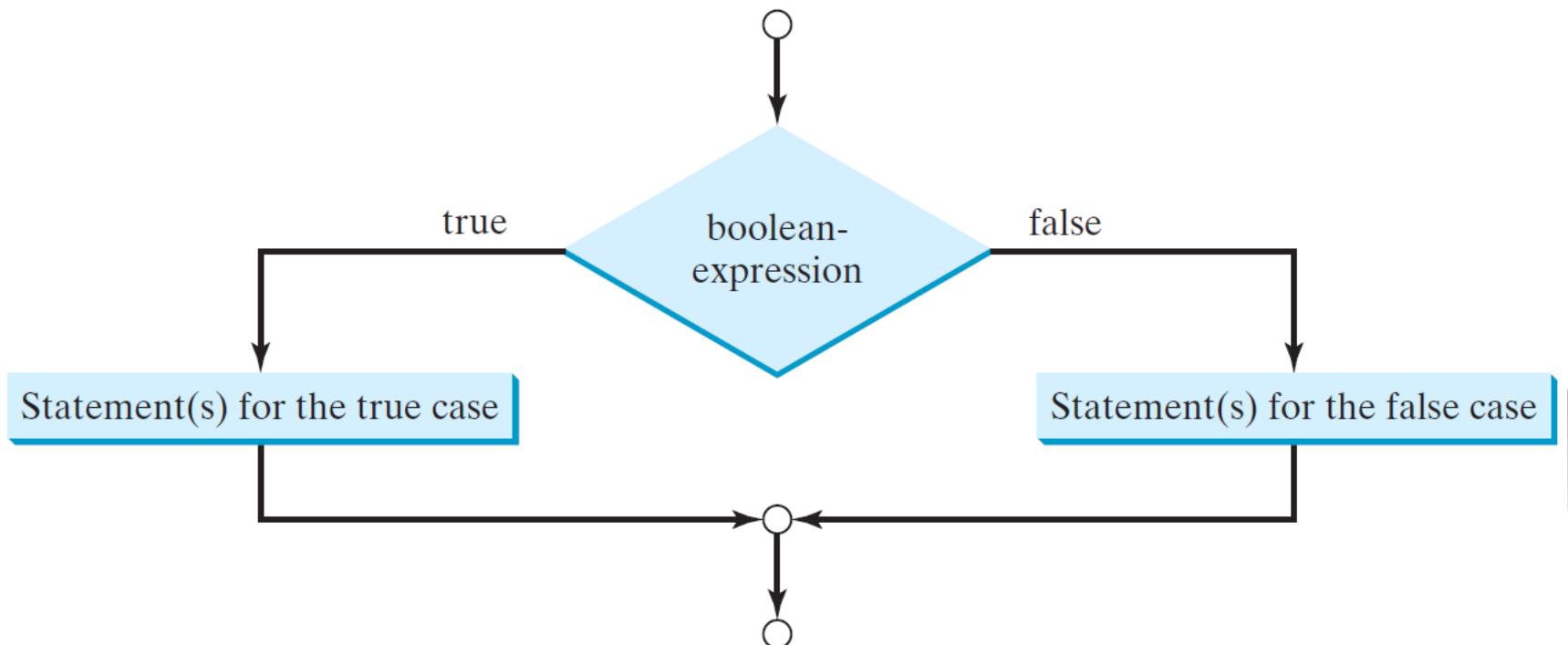
Ví dụ

Viết chương trình yêu cầu người dùng nhập vào một số nguyên. Nếu số nguyên có chia hết cho 5 thì in HiFive. Nếu số nguyên có chia hết cho 2 thì in HiEven.



Câu lệnh if...else

```
if (boolean-expression) {  
    statement(s)-for-the-true-case;  
}  
else {  
    statement(s)-for-the-false-case;  
}
```



Ví dụ if-else

```
if (radius >= 0) {  
    area = radius * radius * 3.14159;  
  
    System.out.println("The area for the "  
        + "circle of radius " + radius +  
        " is " + area);  
}  
else {  
    System.out.println("Negative input");  
}
```



Nhiều câu lệnh if...else

```
if (score >= 90.0)
    System.out.print("A");
else
    if (score >= 80.0)
        System.out.print("B");
    else
        if (score >= 70.0)
            System.out.print("C");
        else
            if (score >= 60.0)
                System.out.print("D");
            else
                System.out.print("F");
```

(a)

Tương đương

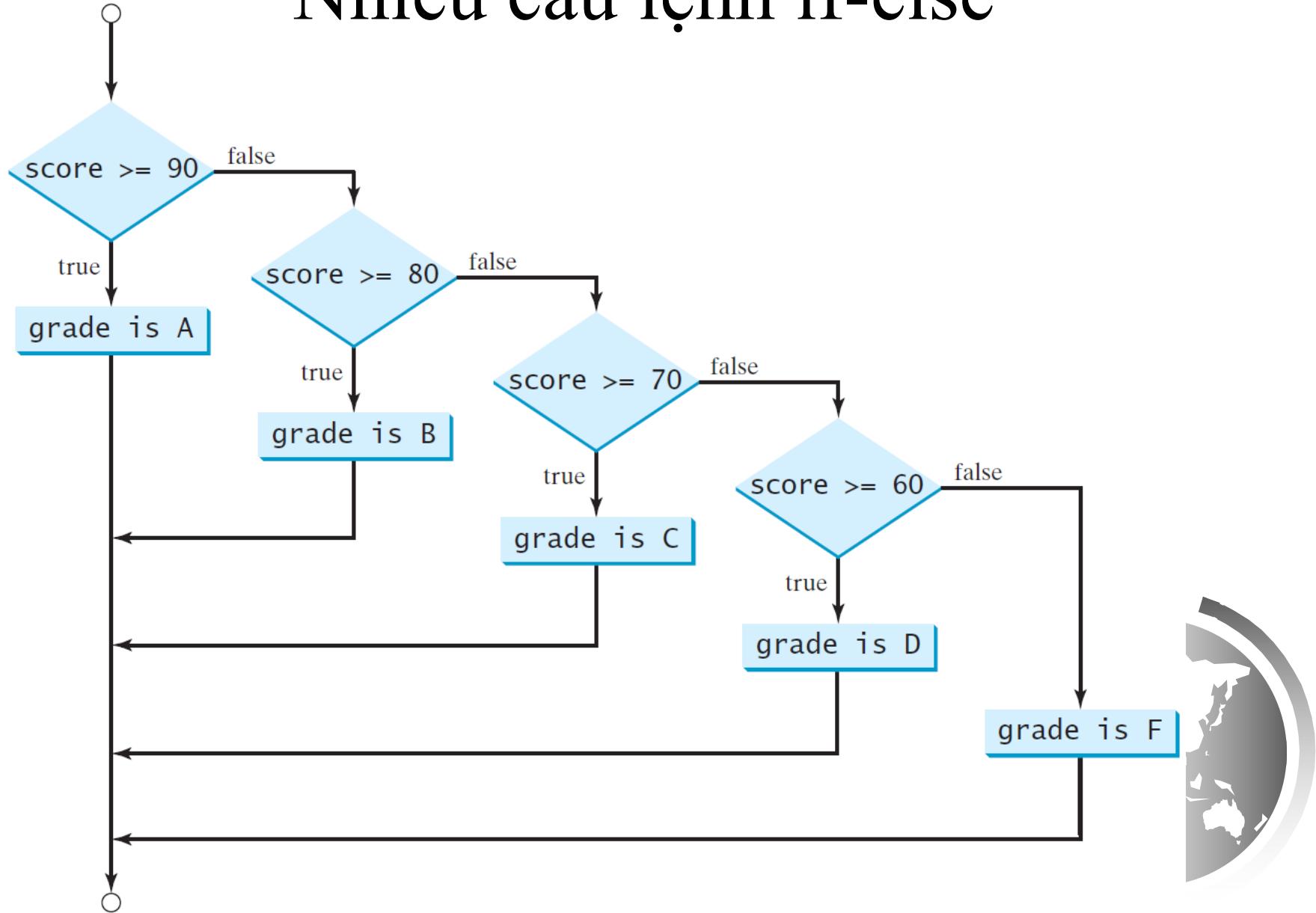
Cách tốt hơn

```
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```

(b)



Nhiều câu lệnh if-else



Chạy từng bước if-else

Giả sử score is 70.0

Điều kiện sai (false)

```
if (score >= 90.0)
```

```
    System.out.print("A");  
else if (score >= 80.0)  
    System.out.print("B");  
else if (score >= 70.0)  
    System.out.print("C");  
else if (score >= 60.0)  
    System.out.print("D");  
else  
    System.out.print("F");
```



Chạy từng bước if-else

Giả sử score is 70.0

Điều kiện sai (false)

```
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```



Chạy từng bước if-else

Giả sử score is 70.0

Điều kiện đúng (true)

```
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```



Chạy từng bước if-else

Giả sử score is 70.0

grade là C

```
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```



Chạy từng bước if-else

Giả sử score is 70.0

Kết thúc câu lệnh if

```
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```



Lưu ý

Mệnh đề else đi kèm với mệnh đề if gần nhất trong cùng 1 khối lệnh.

```
int i = 1, j = 2, k = 3;  
  
if (i > j)  
    if (i > k)  
        System.out.println("A");  
else  
    System.out.println("B");
```

(a)

Equivalent

This is better
with correct
indentation

```
int i = 1, j = 2, k = 3;  
  
if (i > j)  
    if (i > k)  
        System.out.println("A");  
    else  
        System.out.println("B");
```

(b)



Lưu ý

Để mệnh đề else khớp với mệnh đề if đầu tiên, cần thêm cặp dấu ngoặc nhọn:

```
int i = 1;  
int j = 2;  
int k = 3;  
if (i > j) {  
    if (i > k)  
        System.out.println("A");  
}  
else  
    System.out.println("B");
```



Một số lỗi cơ bản

Thêm dấu chấm phẩy vào cuối mệnh đề if.

```
if (radius >= 0);  
{  
    ← Sai  
    area = radius*radius*PI;  
    System.out.println(  
        "The area for the circle of radius " +  
        radius + " is " + area);  
}
```

Lỗi này khó phát hiện (Lỗi logic).

Lỗi này thường xuất hiện khi dùng next-line block style.



MEO

```
if (number % 2 == 0)
    even = true;
else
    even = false;
```

(a)

Equivalent

```
boolean even
= number % 2 == 0;
```

(b)



MEO

```
if (even == true)  
    System.out.println(  
        "It is even.");
```

(a)

Equivalent

```
if (even)  
    System.out.println(  
        "It is even.");
```

(b)



Problem: An Improved Math Learning Tool

This example creates a program to teach a first grade child how to learn subtractions. The program randomly generates two single-digit integers number1 and number2 with number1 >= number2 and displays a question such as “What is $9 - 2?$ ” to the student. After the student types the answer, the program displays whether the answer is correct.



SubtractionQuiz

Run

Problem: Body Mass Index

Body Mass Index (BMI) is a measure of health on weight. It can be calculated by taking your weight in kilograms and dividing by the square of your height in meters. The interpretation of BMI for people 16 years or older is as follows:

BMI	Interpretation
$\text{BMI} < 18.5$	Underweight
$18.5 \leq \text{BMI} < 25.0$	Normal
$25.0 \leq \text{BMI} < 30.0$	Overweight
$30.0 \leq \text{BMI}$	Obese



[ComputeAndInterpretBMI](#)

Run



Problem: Computing Taxes

The US federal personal income tax is calculated based on the filing status and taxable income. There are four filing statuses: single filers, married filing jointly, married filing separately, and head of household. The tax rates for 2009 are shown below.

Marginal Tax Rate	Single	Married Filing Jointly or Qualifying Widow(er)	Married Filing Separately	Head of Household
10%	\$0 – \$8,350	\$0 – \$16,700	\$0 – \$8,350	\$0 – \$11,950
15%	\$8,351 – \$33,950	\$16,701 – \$67,900	\$8,351 – \$33,950	\$11,951 – \$45,500
25%	\$33,951 – \$82,250	\$67,901 – \$137,050	\$33,951 – \$68,525	\$45,501 – \$117,450
28%	\$82,251 – \$171,550	\$137,051 – \$208,850	\$68,526 – \$104,425	\$117,451 – \$190,200
33%	\$171,551 – \$372,950	\$208,851 – \$372,950	\$104,426 – \$186,475	\$190,201 – \$372,950
35%	\$372,951+	\$372,951+	\$186,476+	\$372,951+

Problem: Computing Taxes, cont.

```
if (status == 0) {  
    // Compute tax for single filers  
}  
  
else if (status == 1) {  
    // Compute tax for married file jointly  
    // or qualifying widow(er)  
}  
  
else if (status == 2) {  
    // Compute tax for married file separately  
}  
  
else if (status == 3) {  
    // Compute tax for head of household  
}  
  
else {  
    // Display wrong status  
}
```

ComputeTax

Run



Toán tử luận lý

Toán tử	Tên	Description
!	not	logical negation
&&	and	logical conjunction
	or	logical disjunction
^	exclusive or	logical exclusion

Toán tử NOT

p	$\neg p$	Ví dụ (giả sử age = 24, weight = 140)
true	false	$\neg(\text{age} > 18) \implies \text{false}$, vì $(\text{age} > 18)$ là true.
false	true	$\neg(\text{weight} == 150) \implies \text{true}$, vì $(\text{weight} == 150)$ là false.

Toán tử AND

p_1	p_2	$p_1 \&& p_2$	Ví dụ (giả sử age = 24, weight = 140)
false	false	false	$(age \leq 18) \&& (weight < 140) \Rightarrow \text{false}$, vì $(age > 18)$ và $(weight \leq 140)$ đều là false.
false	true	false	
true	false	false	$(age > 18) \&& (weight > 140) \Rightarrow \text{false}$, vì $(weight > 140)$ là false.
true	true	true	$(age > 18) \&& (weight \geq 140) \Rightarrow \text{true}$, vì cả $(age > 18)$ và $(weight \geq 140)$ là true.

Toán tử OR

p_1	p_2	$p_1 \parallel p_2$	Ví dụ (giả sử age = 24, weight = 140)
false	false	false	
false	true	true	$(age > 34) \parallel (weight \leq 140) \implies \text{true}$, vì $(age > 34)$ là false, nhưng $(weight \leq 140)$ là true.
true	false	true	$(age > 14) \parallel (weight \geq 150) \implies \text{false}$, vì $(age > 14)$ là true.
true	true	true	

Toán tử XOR

p_1	p_2	$p_1 \wedge p_2$	Ví dụ (giả sử age = 24, weight = 140)
false	false	false	$(age > 34) \wedge (weight > 140) \implies \text{true}$, vì $(age > 34)$ là false và $(weight > 140)$ là false.
false	true	true	$(age > 34) \wedge (weight \geq 140) \implies \text{true}$, vì $(age > 34)$ là false nhưng $(weight \geq 140)$ là true.
true	false	true	$(age > 14) \wedge (weight > 140) \implies \text{true}$, vì $(age > 14)$ là true và $(weight > 140)$ là false.
true	true	false	

Ví dụ

Viết chương trình kiểm tra xem một số nguyên được nhập từ bàn phím có **chia hết cho 2 và 3** hay không và có **chia hết cho 2 hoặc chia hết cho 3** hay không?



Ví dụ

```
System.out.println("Is " + number + " divisible by 2 and 3? " +  
((number % 2 == 0) && (number % 3 == 0)));
```

```
System.out.println("Is " + number + " divisible by 2 or 3? " +  
((number % 2 == 0) || (number % 3 == 0)));
```

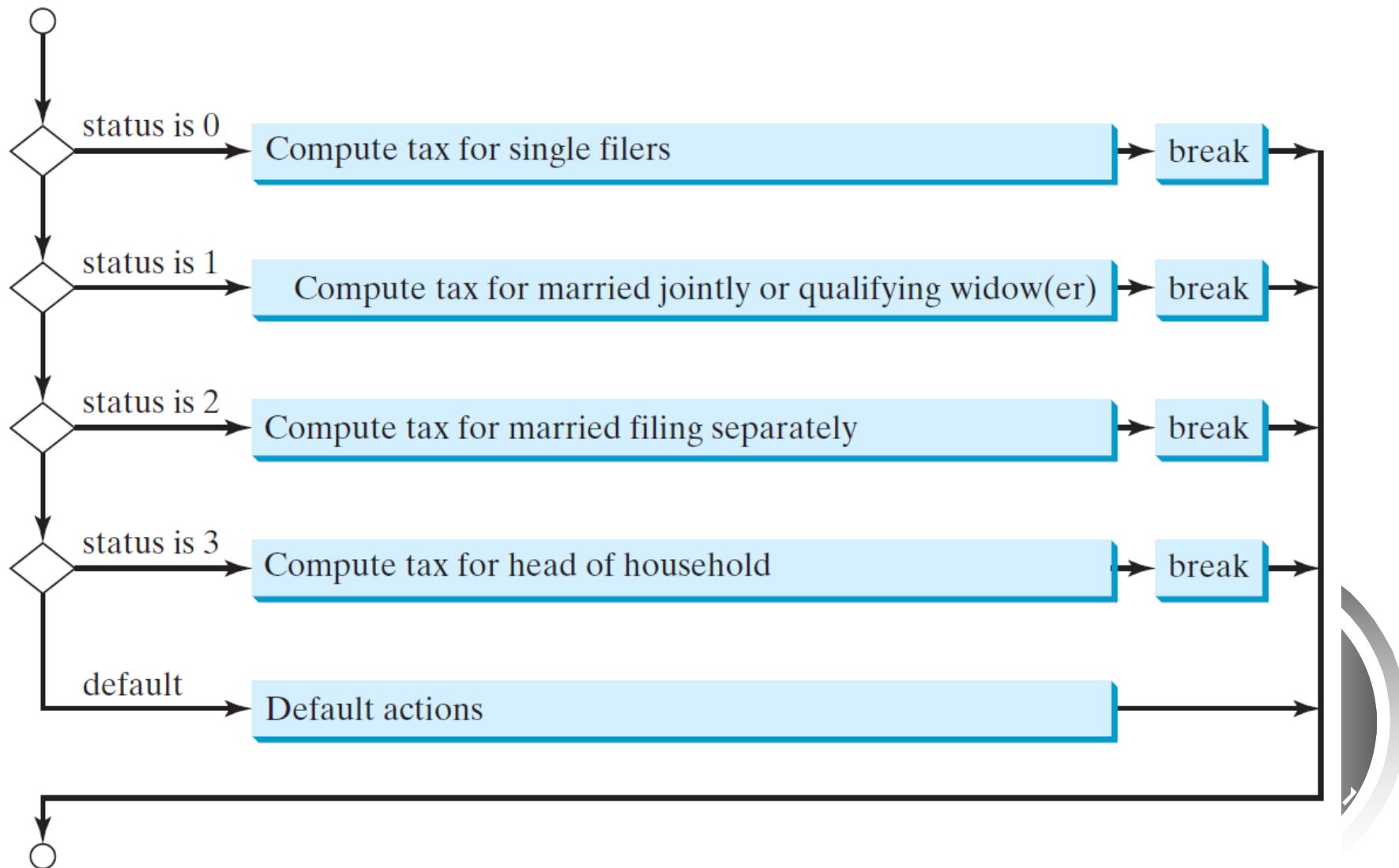


Câu lệnh switch

```
switch (status) {  
    case 0: compute taxes for single filers;  
        break;  
    case 1: compute taxes for married file jointly;  
        break;  
    case 2: compute taxes for married file separately;  
        break;  
    case 3: compute taxes for head of household;  
        break;  
    default: System.out.println("Errors: invalid status");  
        System.exit(1);  
}
```



Lưu đồ câu lệnh switch



Luật của câu lệnh switch

switch-expression là một trong những kiểu giá trị: char, byte, short, hoặc int.

value1, ..., và valueN phải cùng kiểu dữ liệu với switch-expression. Nếu giá trị của case bằng giá trị của switch-expression thì sẽ thực hiện các câu lệnh sau case tương ứng đó. Lưu ý: value1, ..., và valueN là các biểu thức hằng (nghĩa là không chứa biến trong biểu thức, vd: $1 + x$).

```
switch (switch-expression) {  
    case value1: statement(s)1;  
    break;  
    case value2: statement(s)2;  
    break;  
    ...  
    case valueN: statement(s)N;  
    break;  
    default: statement(s)-for-default;  
}
```



Luật của câu lệnh switch

Từ khoá break là không bắt buộc.

Nếu không có break thì câu lệnh sau case kế tiếp sẽ được thực thi.

Trường hợp default là không bắt buộc, có thể được sử dụng để thực hiện các hành động khi các case cụ thể không khớp với switch-expression.

```
switch (switch-expression) {  
    case value1: statement(s)1;  
    break;  
    case value2: statement(s)2;  
    break;  
    ...  
    case valueN: statement(s)N;  
    break;  
    default: statement(s)-for-default;  
}
```



Chạy từng bước câu lệnh switch

Giả sử day = 2:

```
switch (day) {  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
    case 5: System.out.println("Weekday"); break;  
    case 0:  
    case 6: System.out.println("Weekend");  
}
```



Chạy từng bước câu lệnh switch

Trùng với case 2

```
switch (day) {  
    case 1:  
        case 2:  
        case 3:  
        case 4:  
        case 5: System.out.println("Weekday"); break;  
    case 0:  
        case 6: System.out.println("Weekend");  
}
```



Chạy từng bước câu lệnh switch

Chuyển qua case 3

```
switch (day) {  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
    case 5: System.out.println("Weekday"); break;  
    case 0:  
    case 6: System.out.println("Weekend");  
}
```



Chạy từng bước câu lệnh switch

Chuyển qua case 4

```
switch (day) {  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
        System.out.println("Weekday"); break;  
    case 0:  
    case 6: System.out.println("Weekend");  
}
```



Chạy từng bước câu lệnh switch

Chuyển qua case 5

```
switch (day) {  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
    case 5: System.out.println("Weekday"); break;  
    case 0:  
    case 6: System.out.println("Weekend");  
}
```



Chạy từng bước câu lệnh switch

Gặp break

```
switch (day) {  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
    case 5: System.out.println("Weekday"); break;  
    case 0:  
    case 6: System.out.println("Weekend");  
}
```



Chạy từng bước câu lệnh switch

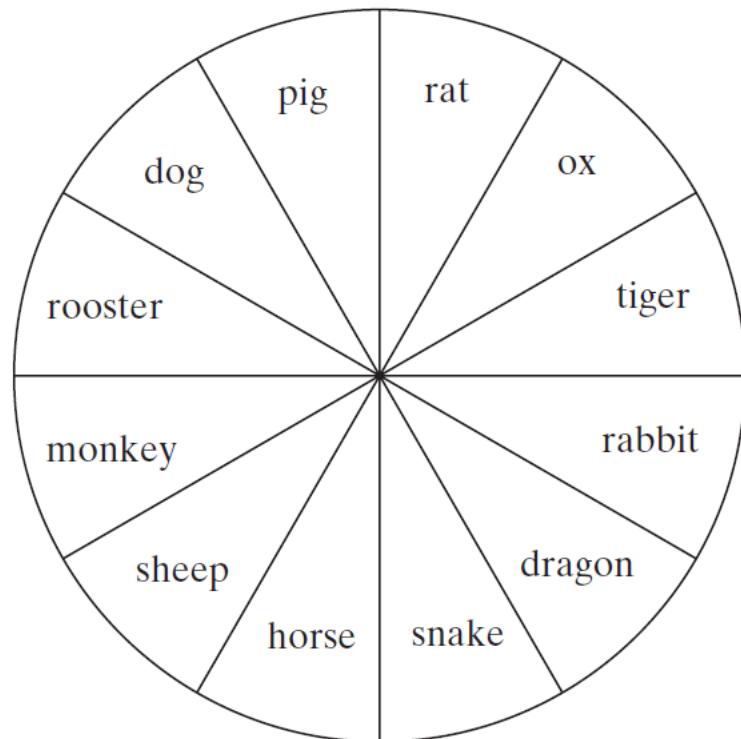
Thoát câu lệnh switch

```
switch (day) {  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
        System.out.println("Weekday"); break;  
    case 0:  
    case 6: System.out.println("Weekend");  
}
```



Ví dụ: 12 con giáp

Viết chương trình yêu cầu người dùng nhập vào một năm và hiển thị con giáp của năm đó.



year % 12 = {

0: monkey
1: rooster
2: dog
3: pig
4: rat
5: ox
6: tiger
7: rabbit
8: dragon
9: snake
10: horse
11: sheep



Toán tử điều kiện

```
if (x > 0)
```

```
    y = 1
```

```
else
```

```
    y = -1;
```

Tương đương với

```
y = (x > 0) ? 1 : -1;
```

```
(boolean-expression) ? expression1 : expression2
```



Toán tử điều kiện

```
if (num % 2 == 0)
    System.out.println(num + "is even");
else
    System.out.println(num + "is odd");
```

```
System.out.println(
    (num % 2 == 0)? num + "is even" :
    num + "is odd");
```



Toán tử điều kiện

boolean-expression ? exp1 : exp2



Ví dụ

In 100 chuỗi “Welcome to Java!”:

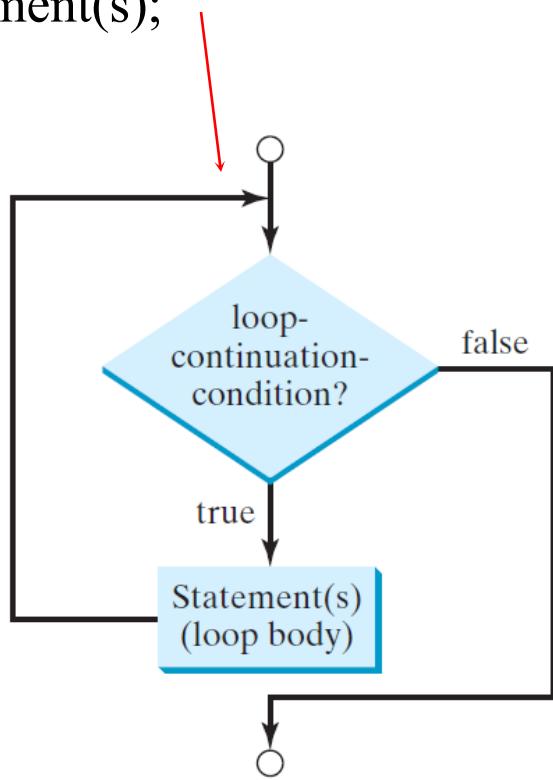
100
times

```
System.out.println("Welcome to Java!");
...
...
...
System.out.println("Welcome to Java!");
System.out.println("Welcome to Java!");
System.out.println("Welcome to Java!");
```

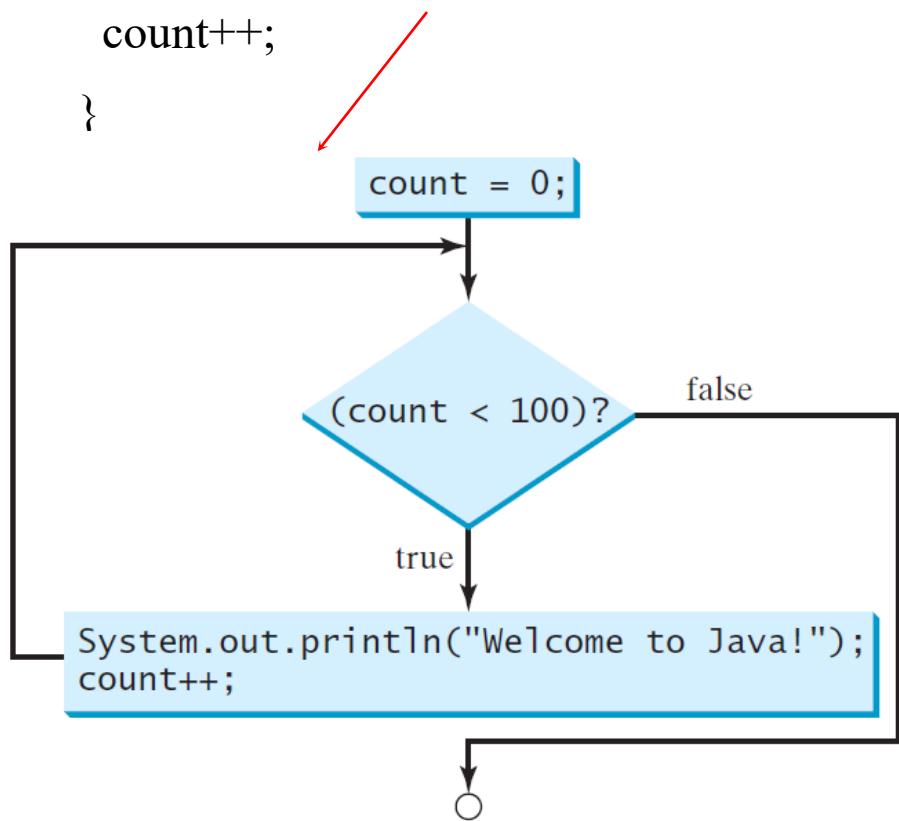


Vòng lặp while

```
while (loop-continuation-condition) {  
    // loop-body;  
    Statement(s);  
}
```



```
int count = 0;  
while (count < 100) {  
    System.out.println("Welcome to Java!");  
    count++;  
}
```



Chạy từng bước vòng lặp while

```
int count = 0;
```

Khởi tạo count

```
while (count < 2) {  
    System.out.println("Welcome to Java!");  
    count++;  
}
```



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```

(count < 2) => true



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```

In Welcome to Java



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```

Tăng count lên 1 đơn vị
==> count = 1



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```

(count < 2) => true vì count = 1



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```

In Welcome to Java



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```

Tăng count lên 1 đơn vị
==> count = 2



Chạy từng bước vòng lặp while

```
int count = 0;
```

```
while (count < 2) {
```

(count < 2) ==> false vì count = 2

```
    System.out.println("Welcome to Java!");
```

```
    count++;
```

```
}
```



Chạy từng bước vòng lặp while

```
int count = 0;  
while (count < 2) {  
    System.out.println("Welcome to Java!");  
    count++;  
}
```

Thoát khỏi vòng lặp. Thực hiện câu lệnh sau vòng lặp.



Lưu ý

Không sử dụng giá trị số chấm động để kiểm tra bằng trong việc điều khiển vòng lặp. Vì số chấm động là các giá trị xấp xỉ.

Tính biểu thức sau $1 + 0.9 + 0.8 + \dots + 0.1$

```
double item = 1; double sum = 0;
```

```
while (item != 0) { // Không chắc chắn item sẽ bằng 0
```

```
    sum += item;
```

```
    item -= 0.1;
```

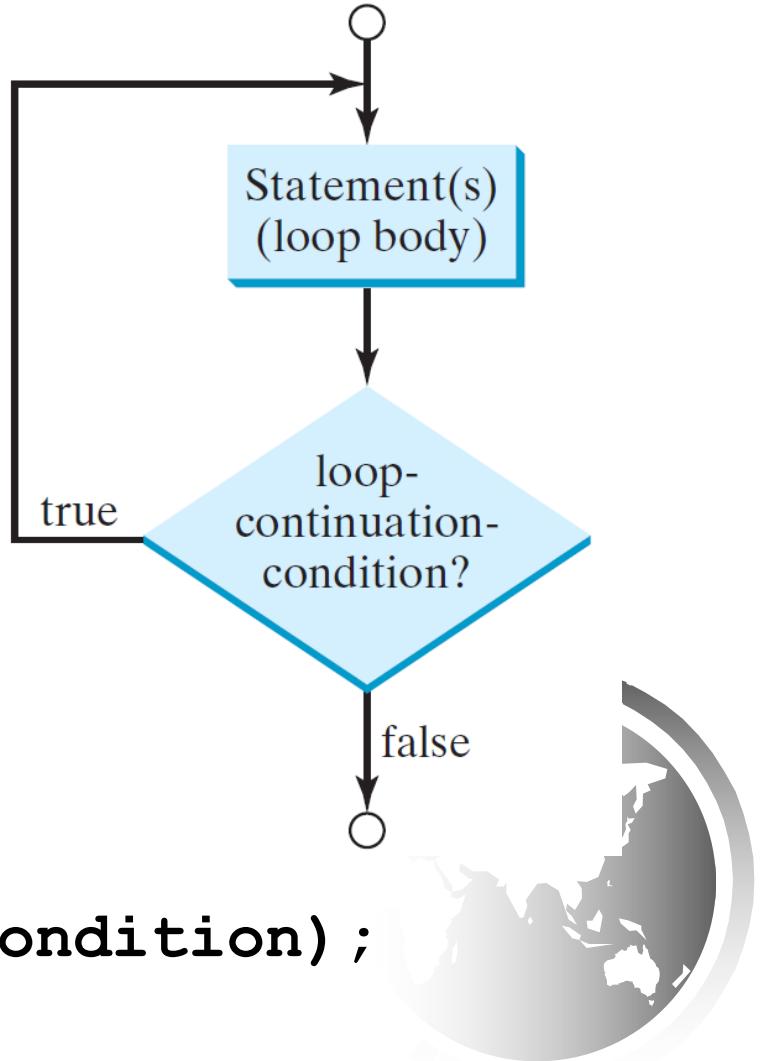
```
}
```

```
System.out.println(sum);
```



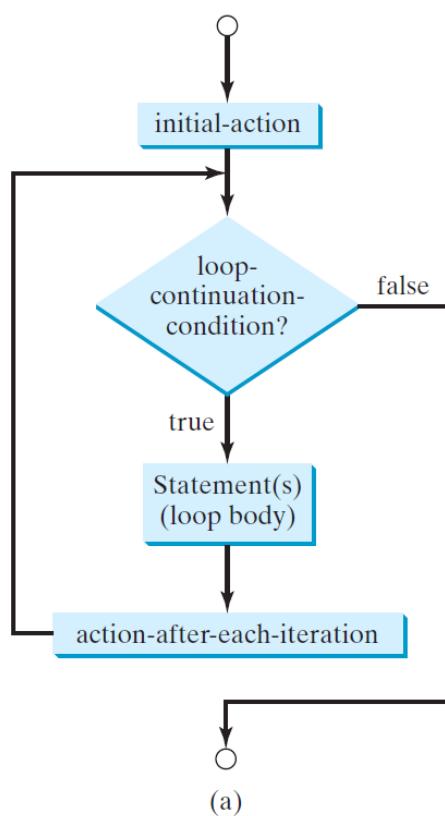
Vòng lặp do-while

```
do {  
    // Loop body;  
    Statement(s);  
} while (loop-continuation-condition);
```

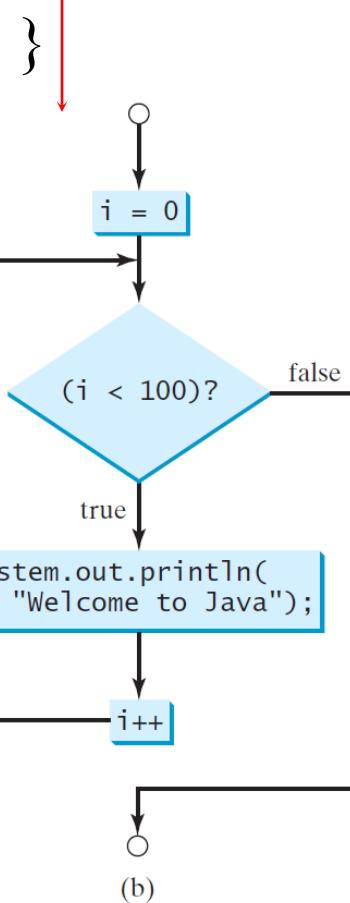


Vòng lặp for

```
for (initial-action; loop-
    continuation-condition; action-
    after-each-iteration) {
    // loop body;
    Statement(s);
}
```



```
int i;
for (i = 0; i < 100; i++) {
    System.out.println(
        "Welcome to Java!");
}
```



c. All



Chạy từng bước vòng lặp for

```
int i;
```

```
for (i = 0; i < 2; i++) {  
    System.out.println(  
        "Welcome to Java!");  
}
```

Khai báo i



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println(  
        "Welcome to Java!");  
}
```

Khởi tạo i
==> i = 0



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println( "Welcome to Java!");  
}
```

$(i < 2) \Rightarrow \text{true}$
vì $i = 0$



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!");  
}
```

In Welcome to Java



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!");  
}
```

Tăng i lên 1 đơn vị
==> i = 1



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!");  
}
```

$(i < 2) \Rightarrow \text{true}$
vì $i = 1$



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!");  
}
```

In Welcome to Java



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!");  
}
```

Tăng i lên 1 đơn vị
==> i = 2



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!");  
}
```

$(i < 2) \Rightarrow \text{false}$
vì $i = 2$



Chạy từng bước vòng lặp for

```
int i;  
for (i = 0; i < 2; i++) {  
    System.out.println("Welcome to Java!")  
}
```

Thoát khỏi vòng lặp. Thực hiện câu lệnh sau vòng lặp.



Lưu ý

initial-action và action-after-each-iteration trong vòng lặp for có thể không có hoặc 1 danh sách các biểu thức cách nhau bởi dấu phẩy.

```
for (int i = 1; i < 100; System.out.println(i++));
```

```
for (int i = 0, j = 0; (i + j < 10); i++, j++) {
```

```
    // Do something
```

```
}
```



Lưu ý

Nếu loop-continuation-condition trong vòng lặp for bị thiếu thì ngầm hiểu là true.

```
for ( ; ; ) {  
    // Do something  
}
```

Equivalent

```
while (true) {  
    // Do something  
}
```

(a)

(b)



Lưu ý

Việc thêm dấu chấm phẩy vào cuối mệnh đề for là một lỗi phổ biến:

```
for (int i=0; i<10; i++) ;  
{  
    System.out.println("i is " + i);  
}
```

Logic
Error



Lưu ý

Tương tự với vòng lặp while:

```
int i=0;  
while (i < 10); ← Logic Error  
{  
    System.out.println("i is " + i);  
    i++;  
}
```

Nhưng với vòng lặp do...while thì dấu chấm phẩy ở sau while là cần thiết.

```
int i=0;  
do {  
    System.out.println("i is " + i);  
    i++;  
} while (i<10); ← Correct
```



Nên sử dụng vòng lặp nào?

Cả 3 vòng lặp while, do-while và for là tương đương nhau

```
while (loop-continuation-condition) {  
    // Loop body  
}
```

(a)

Equivalent

```
for ( ; loop-continuation-condition; ) {  
    // Loop body  
}
```

(b)

```
for (initial-action;  
     loop-continuation-condition;  
     action-after-each-iteration) {  
    // Loop body;  
}
```

(a)

Equivalent

```
initial-action;  
while (loop-continuation-condition) {  
    // Loop body;  
    action-after-each-iteration;  
}
```

(b)

Lời khuyên

Vòng lặp for được sử dụng nếu biết trước số lần lặp, vd: in một thông điệp 100 lần.

Vòng lặp while loop nếu không biết trước số lần lặp, vd: nhập các số nguyên cho đến khi nhập vào số 0 thì dừng.

Vòng lặp do-while có thể được sử dụng thay thế cho vòng lặp while nếu công việc trong vòng lặp cần được thực hiện trước khi kiểm tra điều kiện lặp.



break

```
public class TestBreak {  
    public static void main(String[] args) {  
        int sum = 0;  
        int number = 0;  
  
        while (number < 20) {  
            number++;  
            sum += number;  
            if (sum >= 100)  
                break;  
        }  
        System.out.println("The number is " + number);  
        System.out.println("The sum is " + sum);  
    }  
}
```



continue

```
public class TestContinue {  
    public static void main(String[] args) {  
        int sum = 0;  
        int number = 0;  
  
        while (number < 20) {  
            number++;  
            if (number == 10 || number == 11)  
                continue;  
            sum += number;  
        }  
        System.out.println("The sum is " + sum);  
    }  
}
```

