03\_01

List six relational operators.

<, <=, ==, !=, >, >=

03\_02

Assuming that x is 1, show the result of the following Boolean expressions:

x > 0

x < 0

x != 0

x >= 0

x != 1

true

false

true

true

false

03\_03

boolean b = true;

i = (int)b;

int i = 1;

boolean b = (boolean)i;

No. Boolean values cannot be cast to other types.

03\_04

Write an if statement that assigns 1 to x if y is greater than 0.

if (y > 0)

x = 1;

03\_05

Write an if statement that increases pay by 3% if score is greater than 90.

if (score > 90)

pay \*= 1.03;

03\_06

What is wrong in the following code?

if radius >= 0

{

area = radius \* radius \* PI;

System.out.println("The area for the circle of " +

" radius " + radius + " is " + area);

}

The parentheses is reuqired for the conidition radius >= 0.

03\_07

Write an if statement that increases pay by 3% if score is greater than 90, otherwise increases pay by 1%.

if (score > 90)

pay \*= 1.03;

else

pay \*= 1.01;

03\_08

if (number % 2 == 0)

System.out.println(number + " is even.");

else

System.out.println(number + " is odd.");

If number is 30, (a) displays

30 is even

30 is odd

(b) displays

30 is even

If number is 35, (a) displays

35 is odd

(b) displays

35 is odd

03\_09

Suppose x = 3 and y = 2; show the output, if any, of the following code. What is the output if x = 3 and y = 4? What is the output if x = 2 and y = 2? Draw a flowchart of the code.

**if** (x > 2) {

**if** (y > 2) {

z = x + y;

System.out.println("z is " + z);

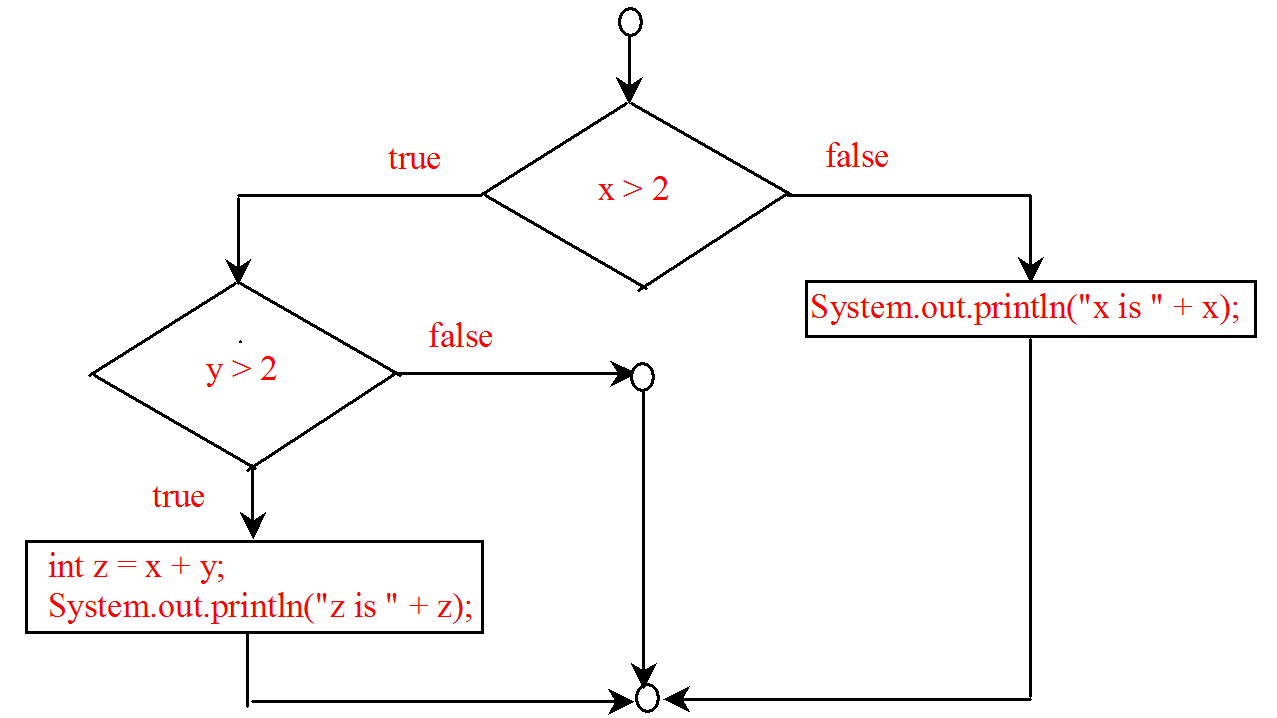
}

}

**else**

System.out.println("x is " + x);

Note: else matches the first if clause. No output if x = 3 and y = 2. Output is "z is 7" if if x = 3 and y = 4. Output is "x is 2" if if x = 2 and y = 2.



03\_10

**if** (score >= 60.0)

System.out.println("D");

**else** **if** (score >= 70.0)

System.out.println("C");

**else** **if** (score >= 80.0)

System.out.println("B");

**else** **if** (score >= 90.0)

System.out.println("A");

**else**

System.out.println("F");

Consider score 90, what will be the grade? It will be D.

03\_11

Which of the following statements are equivalent? Which ones are correctly indented?

(a)

**if** (i > 0) **if**

(j > 0)

x = 0; **else**

**if** (k > 0) y = 0;

**else** z = 0;

(b)

**if** (i > 0) {

**if** (j > 0)

x = 0;

**else** **if** (k > 0)

y = 0;

}

**else**

z = 0;

(c)

**if** (i > 0)

**if** (j > 0)

x = 0;

**else** **if** (k > 0)

y = 0;

**else**

z = 0;

(d)

**if** (i > 0)

**if** (j > 0)

x = 0;

**else** **if** (k > 0)

y = 0;

**else**

z = 0;

a, c, and d are the same. (B) and (C) are correctly indented.

03\_12

Rewrite the following statement using a Boolean expression:

**if** (count % 10 == 0)

newLine = **true**;

**else**

newLine = **false**;

newLine = (count % 10 == 0);

03\_13

Are the following statements correct? Which one is better?

(a)

**if** (age < 16)

System.out.println

("Cannot get a driver's license");

**if** (age >= 16)

System.out.println

("Can get a driver's license");

(b)

**if** (age < 16)

System.out.println

("Cannot get a driver's license");

**else**

System.out.println

("Can get a driver's license");

Both are correct. (b) is better.

03\_14

What is the output of the following code if number is 14, 15, or 30?

(a)

**if** (number % 2 == 0)

System.out.println

(number + " is even");

**if** (number % 5 == 0)

System.out.println

(number + " is multiple of 5");

(b)

**if** (number % 2 == 0)

System.out.println

(number + " is even");

**else** **if** (number % 5 == 0)

System.out.println

(number + " is multiple of 5");

For (a) if number is 14, the output is

14 is even

if number is 15, the output is

15 is multiple of 5

if number is 30, the output is

30 is even

30 is multiple of 5

For (b) if number is 14, the output is

14 is even

If number is 15, the output is

15 is multiple of 5

if number is 30, the output is

30 is even

03\_14

Which of the following is a possible output from invoking Math.random()?

323.4, 0.5, 34, 1.0, 0.0, 0.234

0.5, 0.0, 0.234

03\_15

a. How do you generate a random integer i such that 0 <= i < 20 ?  
b. How do you generate a random integer i such that 10 <= i < 20  
c. How do you generate a random integer i such that 10 <= i <= 50  
d. Write an expression that returns 0 or 1 randomly.

(a) (int)(Math.random() \* 20)  
(b) 10 + (int)(Math.random() \* 10)  
(c) 10 + (int)(Math.random() \* 41)  
(d) (int)(Math.random() \* 2)

03\_16

Are the following two statements equivalent?

(a)

**if** (income <= 10000)

tax = income \* 0.1;

**else** **if** (income <= 20000)

tax = 1000 +

(income - 10000) \* 0.15;

(b)

**if** (income <= 10000)

tax = income \* 0.1;

**else** **if** (income > 10000 &&

income <= 20000)

tax = 1000 +

(income - 10000) \* 0.15;

Yes

03\_17

(a) Write a Boolean expression that evaluates to true if a number stored in variable num is between 1 and 100.  
(b) Write a Boolean expression that evaluates to true if a number stored in variable num is between 1 and 100 or the number is negative.

(a) (num > 1) && (num < 100)  
(b) (num > 1) && (num < 100) || num < 0

03\_18

(a) Write a Boolean expression for |x - 5| < 4.5.  
(b) Write a Boolean expression for |x - 5| > 4.5.

(a) (x - 5) < 4.5 && (x - 5) > -4.5  
(b) (x - 5) > 4.5 || (x - 5) < -4.5

03\_19

Assume that x and y are int type. Which of the following are legal Java expressions?

x > y > 0

x = y && y

x /= y

x or y

x and y

(x != 0) || (x = 0)

x > y > 0 is incorrect  
x = y && y is incorrect  
x /= y is correct  
x or y is incorrect  
x and y is incorrect  
(x != 0) || (x = 0) is incorrect on x = 0. It should be x == 0.

03\_20

Are the following two expressions the same?

(a) x % 2 == 0 && x % 3 == 0

(b) x % 6 == 0

Yes

03\_21

What is the value of the expression x >= 50 && x <= 100 if x is 45, 67, or 101?

If x is 45, the expression is false.  
If x is 67, the expression is true.  
If x is 101, the expression is false.

03\_22

Suppose, when you run the following program, you enter the input 2 3 6 from the console. What is the output?

**public** **class** Test {

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

java.util.Scanner input = **new** java.util.Scanner(System.in);

**double** x = input.nextDouble();

**double** y = input.nextDouble();

**double** z = input.nextDouble();

System.out.println("(x < y && y < z) is " + (x < y && y < z));

System.out.println("(x < y || y < z) is " + (x < y || y < z));

System.out.println("!(x < y) is " + !(x < y));

System.out.println("(x + y < z) is " + (x + y < z));

System.out.println("(x + y > z) is " + (x + y > z));

}

}

(x < y && y < z) is true  
(x < y || y < z) is true  
!(x < y) is false  
(x + y < z) is true  
(x + y > z) is false

03\_23

Write a Boolean expression that evaluates to true if age is greater than 13 and less than 18.

age > 13 && age < 18

03\_24

Write a Boolean expression that evaluates to true if weight is greater than 50 pounds or height is greater than 60 inches.

weight > 50 || height > 60.

03\_25

Write a Boolean expression that evaluates to true if weight is greater than 50 pounds and height is greater than 60 inches.

weight > 50 && height > 60.

03\_26

Write a Boolean expression that evaluates to true if either weight is greater than 50 pounds or height is greater than 60 inches, but not both.

weight > 50 ^ height > 60.

03\_27

Write a Boolean expression that evaluates to true if either weight is greater than 50 pounds or height is greater than 60 inches, but not both.

weight > 50 ^ height > 60.

03\_28

What happens if you enter an integer as 05?

It will be the same as entering 5.

03\_29

What data types are required for a switch variable? If the keyword break is not used after a case is processed, what is the next statement to be executed? Can you convert a switch statement to an equivalent if statement, or vice versa? What are the advantages of using a switch statement?

Switch variables must be of char, byte, short, int, or String types. If a break statement is not used, the next case statement is performed. You can always convert a switch statement to an equivalent if statement, but not an if statement to a switch statement. The use of the switch statement can improve readability of the program in some cases. The compiled code for the switch statement is also more efficient than its corresponding if statement.

03\_30

What is y after the following switch statement is executed? Rewrite the code using an if-else statement.

x = 3; y = 3;

**switch** (x + 3) {

**case** 6: y = 1;

default: y += 1;

}

y is 2.

x = 3; y = 3;

if (x + 3 == 6) {

y = 1;

}

y += 1;

03\_31

What is x after the following if-else statement is executed? Use a switch statement to rewrite it and draw the flowchart for the new switch statement.

**int** x = 1, a = 3;

**if** (a == 1)

x += 5;

**else** **if** (a == 2)

x += 10;

**else** **if** (a == 3)

x += 16;

**else** **if** (a == 4)

x += 34;

x is 17

switch (a) {

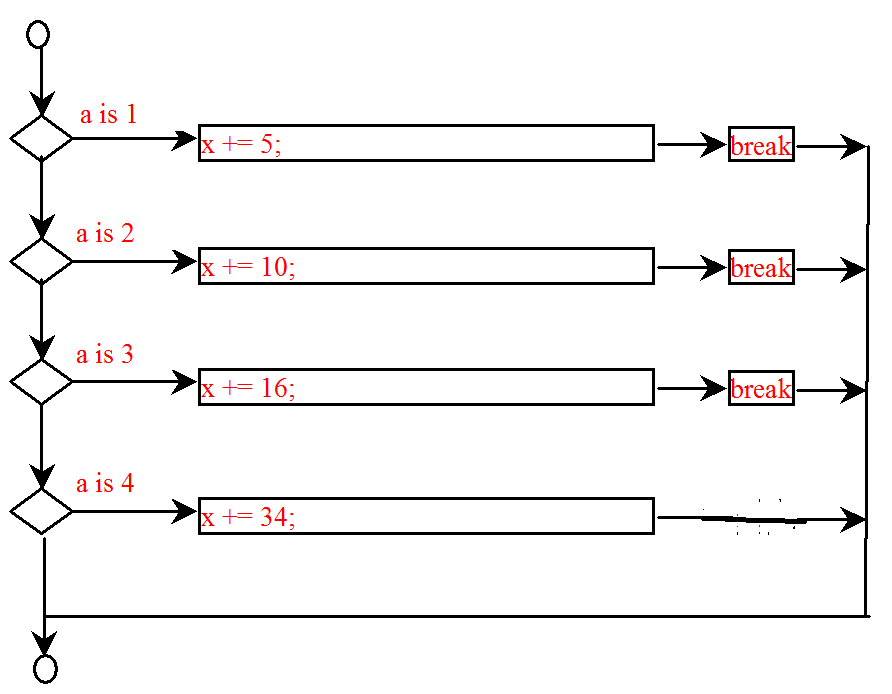
case 1: x += 5; break;

case 2: x += 10; break;

case 3: x += 16; break;

case 4: x += 34;

}



03\_32

Write a switch statement that displays Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, if day is 0, 1, 2, 3, 4, 5, 6, accordingly.

switch (day) {

case 0: System.out.println("Sunday"); break;

case 1: System.out.println("Monday"); break;

case 2: System.out.println("Tuesday"); break;

case 3: System.out.println("Wednesday"); break;

case 4: System.out.println("Thurday"); break;

case 5: System.out.println("Friday"); break;

case 6: System.out.println("Saturday"); break;

}

03\_33

int remainder = year % 12;

if (remainder == 0)

System.out.println("monkey");

else if (remainder == 1)

System.out.println("rooster");

else if (remainder == 2)

System.out.println("dog");

else if (remainder == 3)

System.out.println("pig");

else if (remainder == 4)

System.out.println("rat");

else if (remainder == 5)

System.out.println("ox");

else if (remainder == 6)

System.out.println("tiger");

else if (remainder == 7)

System.out.println("rabbit");

else if (remainder == 8)

System.out.println("dragon");

else if (remainder == 9)

System.out.println("snake");

else if (remainder == 10)

System.out.println("horse");

else

System.out.println("sheep");

03\_34

Suppose that, when you run the following program, you enter the input 2 3 6 from the console. What is the output?

**public** **class** Test {

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

java.util.Scanner input = **new** java.util.Scanner(System.in);

**double** x = input.nextDouble();

**double** y = input.nextDouble();

**double** z = input.nextDouble();

System.out.println((x < y && y < z) ?

"sorted" : "not sorted");

}

}

Sorted

03\_35

Rewrite the following if statements using the conditional operator.

**if** (ages >= 16)

ticketPrice = 20;

**else**

ticketPrice = 10;

ticketPrice = (ages >= 16) ? 20 : 10;

03\_36

Rewrite the following conditional expressions using if-else statements.  
a. score = (x > 10) ? 3 \* scale : 4 \* scale;  
b. tax = (income > 10000) ? income \* 0.2 : income \* 0.17 + 1000;  
c. System.out.println((number % 3 == 0) ? i : j);

(a)

if (x > 10)

score = 3 \* scale;

else

score = 4 \* scale;

(b)

if (income > 10000)

tax = income \* 0.2;

else

tax = income \* 0.17 + 1000;

(c)

if (number % 3 == 0)

System.out.println(i);

else

System.out.println(j);

03\_37

Write conditional expression that returns -1 or 1 randomly.

(int)(Math.random() \* 2) == 0 ? -1 : 1;

03\_38

List the precedence order of the Boolean operators. Evaluate the following expressions:

**true** || **true** && **false**

**true** && **true** || **false**

The precedence order for boolean operators is !, ^, &&, and ||  
true || true && false is true  
true && true || false is true

03\_39

List the precedence order of the Boolean operators. Evaluate the following expressions:

**true** || **true** && **false**

**true** && **true** || **false**

The precedence order for boolean operators is !, ^, &&, and ||  
true || true && false is true  
true && true || false is true

03\_40

True or false? All the binary operators except = are left associative.

True

03\_41

Evaluate the following expressions:

2 \* 2 - 3 > 2 && 4 - 2 > 5

2 \* 2 - 3 > 2 || 4 - 2 > 5

Both are false

03\_42

Is x > 0 && x < 10 the same as x > 0 && x < 10?  
Is x > 0 || x < 10 the same as x > 0 || x < 10?  
Is (x > 0 || x < 10) && y < 0 the same as (x > 0 || (x < 10 && y < 0))?

Yes. Yes. Yes.