

1

. . - - .

2 / /201

,

,

(

)

-

,

```
byte b = 12;  
System.out.println(b);
```

```
b = 255;  
System.out.println(b);
```

int
int

long

,

int

promotion (

(

)

```
byte b = 12;  
System.out.println(b);
```

```
short s = b;  
System.out.println(s);
```

```
int i = s;  
System.out.println(i);
```

```
long l = i;  
System.out.println(l);
```

, **promotion (**)

. .

,

1⁰

, , - - ,
, , ,
()
,

coercition

()

(,)

```
int v = 2293759;  
short vv = (short)v;  
System.out.println(v);  
System.out.println(vv);
```

,

.

,

$$(\quad + \quad)^*(\quad + \quad),$$

- `short a,b,c; a=0; b=0; c = a+b;`

```

+                                     int,

```

int

int.

int

short

;

- `short a,b,c; a=0; b=0; c = (short)a+b;`

) (' - -
,

int

byte short int long float double

char<int

- byte+byte \mapsto int+int \mapsto int
- short*int \mapsto int*int \mapsto int

,
10,
(
-)
0
+123, 1 3, -
+

,

(0),

0

,

+

-

+0123, 01 3, -0

,

(0¹₋),

0

+

-

+0 123, 0 1 3, -0 , 0

$(01),$

0

+

2,

+0 111, 0 00101

, _ () ,

int i = 123_456_789;

, , .

,

long

l L

0xBAD_F00D_FEE1_DEADL

long

-0711

,

,

,

,

,

,

+

différentes

deux opérations d'addition

,

long

int

,

—

*

division entière

/

%

(

)

()

, , (.
)
+ ,

121 13 +

121, 13, ,

, ,
 (,
).
 , (.)
 + , 0 ≤

-121 -10 13 +

+

-121 - 13 + -

, (,)

-121/13 -10 13 + , -10

, , ,

-121/13 - *13 + - , - -

, opérateurs de comparaison

- - , - - - ,

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

,

(int, long,

char)

2

2

2

2

$$1\ 3\ *2\ 3$$

$$\mathbf{11100101010}^{*10}\ \mathbf{111001010100}$$

$$1\ 3\ *1\ 2\ 3$$

$$\mathbf{11100101010}^{*10000}\ \mathbf{111001010100000}$$

()

1 3 / 2 1

11100101010/10 1110010101

1 3 / 1 11

11100101010/10000 1110010

-

```
for (int i=0; i<32; i++)  
    System.out.println(1<<i);
```

```
for (int i=0; i<32; i++)  
    System.out.println(-1<<i);
```


-

```
for (int i=0; i<32; i++)  
    System.out.println(-1>>i);
```

```
for (int i=0; i<32; i++)  
    System.out.println(-1>>>i);
```

>> ,

>>> ,

,

,

(

)

,

,

(-)

,

,

,

,

. .

() , 1 , 32, ,

2

(

,

,

)

32

‘ () ’

-

mémoire

0

1

+1

+2

+3

chiffres base 256

0

1

2

3

-

-

-

-

mémoire

0

1

+1

+2

+3

chiffres base 256

3

2

1

0

-

-

-

-

3 , -

, -

- ,

(), -

(), -

- ()

(,)

32-

32-

-

,

,

,

,

,

,

,

,

,

-

-

-

(
)

()

()

,

,

,

,

[illegible]

$$(1 + \frac{1}{2})^2$$

1 ()

0,

-

1

,3 3

(
101,01011

1/ + 1/1 + 1/32)²

$$\left(\frac{1}{2^3} + \frac{1}{2^1} + \frac{1}{2^5} \right) (101,01011)_2$$

$$\frac{(1,0101011 \cdot 10^{10})_2}{2^3}$$

$$10000001_2, 0101011_2 \cdot 10 + 1111111_2$$

$$0 \frac{1000000101010110000000000000000000}{2^3}$$

`Float.intBitsToFloat()`

0,3

,

,

-

0,3-0,2

0,2-0,1

0,3 0, 1,2 0, 0, 1, 1,2 0,

0,0(1001)^ω, 1,(0011)^ω 10⁻¹⁰,
01111111-10 01111101

00111110100110011001100110011010**10** () 0 3

0,2 0, 0, 1, 1,2 0,

0,(0011)^ω, 1,(1001)^ω 10⁻¹¹,
01111100 01111111-11

00111110010011001100110011001101**1** () 0 3

0,1 0,2 0, 0, 1, 1,2 0

0,0(0011)^ω, 1,(1001)^ω 10⁻¹⁰⁰,
01111011 01111111-100

00111101110011001100110011001101**1** () 0 3

0,3-0,2 - 00111110100110011001100110011010
00111110010011001100110011001101

1001100110011001100110100
- 110011001100110011001101
001100110011001100110011

0011110111001100110011001110

0x3DCCCCCE

0,2-0,1 - 00111110010011001100110011001101
00111101110011001100110011001101

1100110011001100110011010
- 110011001100110011001101
110011001100110011001101

00111101110011001100110011001101

0x3DCCCCCD

0,3-0,2 ' 0x3DCCCCCE

```
public class Essai{  
    public static void main(String argv[]) {  
        int n = 0x12345678;  
        float f = n;  
        int m = (int)f;  
        System.out.println(n);  
        System.out.println(f);  
        System.out.println(m);  
    }  
}
```

```
% java Essai  
305419896  
3.05419904E8  
305419904  
%
```

,
 (0, , -0
 +0)
 , (, 1, 0
 ,)
 (1/0), , 1,
 ≠0
 (,
 0), 0, ≠0,
 0, 2⁻¹²

ε

ε

()

(,

'analyse numérique)

,

.

,

.

```
float r = 0;
while (r!=20) {
    System.out.println(r);
    r = r+0.5f;
}
```

0. 1,

0. 1/ ,

0. 0.2