

第 2 章

2-1 if 文

■ if 文の構文

`if (式) 文` … 式を評価した値が 0 でなければ文を実行。

`if (式) 文1 else 文2` … 式を評価した値が 0 でなければ文 1 を、
そうでなければ文 2 を実行。

上記の構文にそぐわないものは、エラーとなることに注意しよう。

○ `if (a < 3) c = 5;`

○ `if (a == 5) z = 3; else z = 0;`

× `if (a == 5) else z = 0;` // 文 1 が不在

× `if a == 3 c = 5;` // 式を囲む () が不在

■ 評価

C++ の式は原則として、その値を評価することができる。

■ 条件判断等に用いられる演算子

■ 関係演算子

二つのオペランドの大小関係を調べる演算子。

$x < y$	x が y より小さければ 1、そうでなければ 0。
$x > y$	x が y より大きければ 1、そうでなければ 0。
$x \leq y$	x が y 以下 (等しいか小さい) であれば 1、そうでなければ 0。
$x \geq y$	x が y 以上 (等しいか大きい) であれば 1、そうでなければ 0。

※ 注意 ※

! $<=$ 演算子および $>=$ 演算子は、不等号 $<$ あるいは $>$ と $=$ の間に空白を入れてはならない。

! $=$ は必ず不等号の右側である。

$< =$ … 正しい $< =$ … 間違い $= <$ … 間違い

■ 等値演算子

二つのオペランドの等値関係を調べる演算子。

$x = y$	x と y が等しければ 1、そうでなければ 0。
$x != y$	x と y が等しくなければ 1、そうでなければ 0。

■ 論理演算子

論理演算を行う演算子。

$x \&\& y$	x と y の両方が非 0 であれば 1、そうでなければ 0。
$x \ \ y$	x と y のいずれか一方でも非 0 であれば 1、そうでなければ 0。

■ 論理否定演算子

オペランドが 0 であるかどうかを調べる演算子。

$! x$	x の値が 0 であれば 1、そうでなければ 0。
-------	-----------------------------

■ 演習問題 (1)

◆ $<$, $>$, $<=$, $>=$ は大小関係を調べる演算子であり、(1) 演算子と呼ばれる。int 型の変数 a , b , c の値が、それぞれ 10, 10, 20 であるとき、以下の各式を評価した値を示せ。

$a < b \cdots$ (2) $b > c \cdots$ (3)
 $a > c \cdots$ (4) $a <= b \cdots$ (5)
 $b >= c \cdots$ (6) $a >= c \cdots$ (7)

◆ $=$, $!=$ はオペランドが等しいか (等しくないか) を調べる演算子であり、(8) 演算子と呼ばれる。int 型の変数 a , b , c の値が、それぞれ 10, 10, 20 であるとき、以下の各式を評価した値を示せ。

$a == b \cdots$ (9) $b == c \cdots$ (10)
 $a == c \cdots$ (11) $a != b \cdots$ (12)
 $b != c \cdots$ (13) $a != c \cdots$ (14)

◆ $\&\&$, $\|\|$ は論理的な演算を行う演算子であり、(15) 演算子と呼ばれる。int 型の変数 a , b , c の値が、それぞれ 10, 0, 15 であるとき、以下の各式を評価した値を示せ。

$a \&\& b \cdots$ (16) $b \&\& c \cdots$ (17)
 $a \&\& c \cdots$ (18) $a \|\| b \cdots$ (19)
 $b \|\| c \cdots$ (20) $a \|\| c \cdots$ (21)

◆ $!$ はオペランドが 0 であるかどうかを調べる演算子であり、(22) 演算子と呼ばれる。int 型の変数 a , b の値が、それぞれ 8, 0 であるとき、以下の各式を評価した値を示せ。

$!a \cdots$ (23) $!b \cdots$ (24)

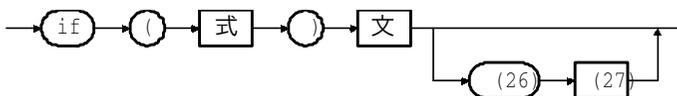
◆ x の値が 3 であるとき、以下に示す if 文の実行によって画面に表示されるのは (25) である。

```

if (x)
    cout << "★\n" ;
else
    cout << "☆\n" ;
    
```

(a) ★
 選択肢
 (b) ☆

◆ 以下に示すのは、if 文の構文図である。



(1)	
(2)	
(3)	
(4)	
(5)	
(6)	
(7)	
(8)	
(9)	
(10)	
(11)	
(12)	
(13)	
(14)	
(15)	
(16)	
(17)	
(18)	
(19)	
(20)	
(21)	
(22)	
(23)	
(24)	
(25)	
(26)	
(27)	

■ 演習問題 (2)

◆ 以下に示すプログラムは、二つの int 型の変数 a, b に整数値を読み込んで、b が a の約数であれば、「b は a の約数です。」と表示し、そうでなければ「b は a の約数ではありません。」と表示するプログラムである。

```
#include <(1)>

int (2) ((3))
{
    int a (4) b; // a, b は整数型オブジェクト (変数)

    (5) << " a の値を入力してください：";
    (6) >> a;
    (5) << " b の値を入力してください：";
    (6) >> b;

    if (a (7) b)
        (5) << " b は a の約数ではありません。¥n";
    else
        (5) << " b は a の約数です。¥n";

    (8) (0);
}
```

(1)	
(2)	
(3)	
(4)	
(5)	
(6)	
(7)	
(8)	
(9)	
(10)	
(11)	

◆ 以下に示すのは、int 型変数 x の値が、偶数であれば「偶数です。」と、奇数であれば「奇数です。」と表示するプログラムの部分である。

```
if ( ((9)) == 0)
    cout << "偶数です。¥n";
else
    cout << "奇数です。¥n";
```

◆ 以下に示すプログラム部分は、いずれも、変数 a と b の値が等しければ「a と b は等しいです。」と、そうでなければ「a と b は等しくありません。」と表示するものである。

《実現例 1》

```
if (a (10) b)
    cout << " a と b は等しいです。¥n";
else
    cout << " a と b は等しくありません。¥n";
```

《実現例 2》

```
if (a (11) b)
    cout << " a と b は等しくありません。¥n";
else
    cout << " a と b は等しいです。¥n";
```

■ 演習問題 (3)

◆ 以下に示すのは、int 型変数 x の値が、正であれば「正の数です。」と、負であれば「負の数です。」と、0 であれば「0 です。」と表示するプログラムの部分である。

```
if ( (1) )
    cout << "正の数です。¥n";
else (2) (3)
    cout << "負の数です。¥n";
else
    cout << "0 です。¥n";
```

◆ 以下に示すのは、int 型変数 a, b の値の差が 1 であれば、「a と b は 1 つ違います。」と表示するプログラムの部分である。

```
if ( (a - b == 1) (4) (b - a == 1) )
    cout << "a と b は 1 つ違います。";
```

◆ 以下に示すのは、int 型変数 a, b が両方とも偶数であれば a, b の和を表示し、そうでなければ a, b の積を表示するプログラムの部分である。

```
if ( (a % 2 == 0) (5) (b % 2 == 0) )
    cout << "a と b の和は" << a + b << " です。¥n";
else
    cout << "a と b の積は" << a * b << " です。¥n";
```

◆ 以下に示すのは、int 型変数 a の値が、正の 1 桁の奇数 (すなわち 1, 3, 5, 7, 9 のいずれか) であれば、「その数は正の 1 桁の奇数です。」と表示するプログラムの部分である。

```
if ( (a (6) 10 == 0) (7) (a % 2) )
    cout << "その数は正の 1 桁の奇数です。¥n";
```

◆ 以下に示すのは、int 型変数 a の下 1 桁の値が、3, 7 のいずれかであれば (たとえば a が 53 とか 857 であれば)、「その数の一番下の桁の値は 3 あるいは 7 ですね。」と表示するプログラムの部分である。

```
b = a (8) 10;
if (b == 3 (9) b == 7)
    cout << "その数の一番下の桁の値は 3 あるいは 7 ですね。¥n";
```

◆ 以下に示すのは、double 型変数 no が 13 の平方根以下であれば、「その値は 13 の平方根以下です。」と表示するプログラムの部分である。

```
if (no * (10) <= (11))
    cout << "その値は 13 の平方根以下です。¥n";
```

(1)	
(2)	
(3)	
(4)	
(5)	
(6)	
(7)	
(8)	
(9)	
(10)	
(11)	

■ 演習問題 (4)

◆ 以下に示すのは、いずれも、int 型変数 x の値が 0 であれば「○」と表示し、そうでなければ「×」と表示するプログラムの部分である。

```
《実現例 1》 if (x)
               cout << "(1) %n";
               else
               cout << "(2) %n";
```

```
《実現例 2》 if (x (3) 0)
               cout << "○%n";
               else
               cout << "×%n";
```

```
《実現例 3》 if (x (4) 0)
               cout << "×%n";
               else
               cout << "○%n";
```

(1)	
(2)	
(3)	
(4)	
(5)	
(6)	
(7)	
(8)	
(9)	
(10)	
(11)	
(12)	

◆ 以下に示すプログラムは、二つの int 型の変数 a, b に整数値を読み込んで、両方とも奇数か、片方が奇数で片方が偶数か、両方とも偶数かを判断するプログラムである。

```
#(5) <iostream.h>

int main(void)
{
    int cnt; // 偶数はいくつか?
    int a, b; // a, bは整数型オブジェクト (変数)

    (6) << " a の値を入力してください：";
    (7) >> a;
    (6) << " b の値を入力してください：";
    (7) >> b;

    cnt = 0;
    if ((8) == 0) cnt = cnt + 1; // aが偶数であれば...
    if ((9) == 0) cnt = cnt + 1; // aが偶数であれば...

    if (cnt == (10))
        (6) << "両方とも奇数です。 %n";
    else if (cnt == (11))
        (6) << "片方が偶数で片方が奇数です。 %n";
    else
        (5) << "両方とも偶数です。 %n";

    (12) (0);
}
```

■ プログラム作成演習 (1)

(2-1) 右に示すようなやり取りによって、二つの int 型の変数 a, b に整数値を読み込んで、b が a の倍数であれば、「b は a の倍数です。」と表示し、そうでなければ「b は a の倍数ではありません。」と表示するプログラムを作成せよ。

整数 a の値を入力して下さい: 7
 整数 b の値を入力して下さい: 14
 b は a の倍数です。

(2-2) 右に示すようなやり取りによって、二つの int 型の変数 a, b に整数値を読み込んで、b が a の 2 乗より小さければ、「b は a の 2 乗より小さいです。」と表示し、そうでなければ「b は a の 2 乗以上です。」と表示するプログラムを作成せよ。

整数 a の値を入力して下さい: 3
 整数 b の値を入力して下さい: 8
 b は a の 2 乗より小さいです。

整数 a の値を入力して下さい: 3
 整数 b の値を入力して下さい: 9
 b は a の 2 乗以上です。

※ a の 2 乗は、 $a * a$ で求めることができる。

(2-3) 右に示すようなやり取りによって、読み込んだ二つの整数値の差を表示するプログラムを作成せよ。

整数を入力して下さい: 3
 整数を入力して下さい: 9
 それらの差は 6 です。

※ 2 つの変数の名前を v1, v2 とする。v1 が v2 より大きければ $v1 - v2$ の値を表示し、そうでなければ $v2 - v1$ の値を表示すればよい。

整数を入力して下さい: 10
 整数を入力して下さい: 9
 それらの差は 1 です。

(2-4) 右に示すようなやり取りによって、読み込んだ二つの実数値の差が 1.0 未満であるかどうかを表示するプログラムを作成せよ。

実数を入力して下さい: 10.5
 実数を入力して下さい: 9.8
 それらの差は 1.0 未満です。

実数を入力して下さい: 11.37
 実数を入力して下さい: 7.8
 それらの差は 1.0 以上です。

(2-5) 右に示すようなやり取りによって、整数値である点数を読み込んで、その評価を表示するプログラムを作成せよ (if 文を用いること)。

点数を入力して下さい: 61
 可です。

点数を入力して下さい: 89
 優です。

評価 : 不可 ... 0 点 ~ 59 点
 可 ... 60 点 ~ 69 点
 良 ... 70 点 ~ 79 点
 優 ... 80 点 ~ 100 点

2-2 switch 文

■ ラベルの出現順序

break 文がないときは、ラベルの出現順序によって実行結果が異なるので注意すること。

```
switch (a) {  
  case 1: cout << "A";  
  case 2: cout << "B";  
}
```

aが1のとき
AB

aが2のとき
B

```
switch (a) {  
  case 2: cout << "B";  
  case 1: cout << "A";  
}
```

aが1のとき
A

aが2のとき
BA

■ 演習問題 (5)

- ◆ 以下に示す switch 文は、a が 1, 2, 3, 4 のとき、それぞれ (1), (2), (3), (4) と出力する。

```
switch (a) {
    case 3: cout << "A";
    case 2: cout << "B"; break;
    default: cout << "C"; break;
}
```

- ◆ int 型の変数 a, b, c の値が、それぞれ 0, 3, 5 であるとき、以下の各式を評価した値を示せ。

(a < b) * (a == 0) ... (5)
 (a == 0) && (b == 0) ... (6)
 (a < b) + (c % 3) ... (7)
 (a != 0) || (b != 0) ... (8)

(1)	
(2)	
(3)	
(4)	
(5)	
(6)	
(7)	
(8)	
(9)	

- ◆ 以下に示すのは、変数 month の値に応じて季節を表示するプログラム部分である。

```
switch (month) {
    case 3: case 4: case 5: cout << "春"; break;
    case 6: case 7: case 8: cout << "夏"; break;
    case 9: case 10: case 11: cout << "秋"; break;
    case 12: case 1: case 2: cout << "冬"; break;
    (9): cout << "そのような月はありません。¥n"; break;
}
```

■ プログラム作成演習 (2)

(2-6) 読み込んだ整数値が 0 であれば「零です。」、1 であれば「壺です。」、2 であれば「弍です。」、それ以外であれば「零、壺、弍以外の数です。」と表示するプログラムを作成せよ。

整数を入力して下さい: 0
零です。

整数を入力して下さい: 2
弍です。

(2-7) 右に示すようなやり取りによって、二つの int 型の変数 a, b に整数値を読み込んで、b が a の倍数であれば、「b は a の倍数です。」と表示し、そうでなければ「b は a の倍数ではありません。」と表示するプログラムを作成せよ。

整数 a の値を入力して下さい: 7
整数 b の値を入力して下さい: 14
b は a の倍数です。