

錬成問題

■ `int` 型の変数 $n1, n2, n3, n4$ の値が、それぞれ 15, 15, 20, 30 であるとする。このとき、以下の各式を評価した値を示せ。

<code>n1 == n2</code> ... <input style="width: 50px;" type="text" value="(1)"/>	<code>n1 == n3</code> ... <input style="width: 50px;" type="text" value="(2)"/>
<code>n1 != n2</code> ... <input style="width: 50px;" type="text" value="(3)"/>	<code>n1 != n4</code> ... <input style="width: 50px;" type="text" value="(4)"/>
<code>n1 >= n2</code> ... <input style="width: 50px;" type="text" value="(5)"/>	<code>n1 > n2</code> ... <input style="width: 50px;" type="text" value="(6)"/>
<code>n2 <= n3</code> ... <input style="width: 50px;" type="text" value="(7)"/>	<code>n2 < n4</code> ... <input style="width: 50px;" type="text" value="(8)"/>
<code>n1 % 5 == 0</code> ... <input style="width: 50px;" type="text" value="(9)"/>	<code>n1 / 5 != 3</code> ... <input style="width: 50px;" type="text" value="(10)"/>
<code>n1 ? n2 : n3</code> ... <input style="width: 50px;" type="text" value="(11)"/>	<code>n1 != n2 ? 5 : 3</code> ... <input style="width: 50px;" type="text" value="(12)"/>
<code>1 && 0</code> ... <input style="width: 50px;" type="text" value="(13)"/>	<code>1 0</code> ... <input style="width: 50px;" type="text" value="(14)"/>
<code>1 && 1</code> ... <input style="width: 50px;" type="text" value="(15)"/>	<code>1 1</code> ... <input style="width: 50px;" type="text" value="(16)"/>
<code>n1 && n2</code> ... <input style="width: 50px;" type="text" value="(17)"/>	<code>n1 n2</code> ... <input style="width: 50px;" type="text" value="(18)"/>
<code>(n1 == n2) && 0</code> ... <input style="width: 50px;" type="text" value="(19)"/>	<code>(n1 == n2) 0</code> ... <input style="width: 50px;" type="text" value="(20)"/>

■ `if` 文の構文図を示せ。

■ 複合文（ブロック）の構文図を示せ。

■ `int` 型の変数 no の値が、1375 であるとする。右に示す `if` 文によって表示されるのは である。

▶ の選択肢 ... (a) ★ (b) ☆

```
if (no)
    puts("★");
else
    puts("☆");
```

■ 以下に示すのは、いずれも、変数 no が 0 であれば「0」と、そうでなければ「非0」と表示するプログラム部分である。

```
if (no)
    puts(" 

```

```
if (no 

```

```
if (no 

```

■ 以下に示すのは、変数 $n1$ が $n2$ の倍数であるかどうかを表示するプログラム部分である。

```
if (

```

■ 以下に示すのは、いずれも、変数 $n1$ と $n2$ が等しければ「それらは等しい。」と、そうでなければ「それらは異なる。」と表示するプログラム部分である。

```
if (n1  n2)
    puts("それらは等しい。");
else
    puts("それらは異なる。");
```

```
if (n1  n2)
    puts("それらは異なる。");
else
    puts("それらは等しい。");
```

■ 以下に示すのは、いずれも、`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部分である。

```
if ((a  0)  (a  10)  (a % 2))
    puts("その数は正の1桁の奇数です。");
```

```
if ((a  1)  (a  10 != )  (a % 2 != )
    puts("その数は正の1桁の奇数です。");
```

■ 以下に示すのは、`int` 型変数 a の下1桁の値が、3か7のいずれか（たとえば53や857など）であれば、「その数の一番下の桁の値は3あるいは7です。」と表示するプログラム部分である。ただし、 a の値は正であるとする。

```
c = a  10;
if (c == 3  c == 7)
    puts("その数の一番下の桁の値は3あるいは7です。");
```

■ 以下に示すのは、読み込んだ整数値が、正、0、負のいずれであるかを表示するプログラムである。

```
#include <.h>

int  (void)
{
    int num;

    printf("整数を入力してください：");
    scanf("%d", &num);

    if ()
        puts("その数は正です。");
    else  ()
        puts("その数は0です。");
    
        puts("その数は負です。");

    return (0);
}
```

(次ページへ続く)

■ 以下に示すのは、読み込んだint型変数aをbで割ったときの商および剰余を表示するプログラム部分である。なお、割り切れるときは商のみを表示する。

```
整数A : 10
整数B : 5
AをBで割ると2です。
```

```
printf("整数A : "); scanf("%d", &a);
printf("整数B : "); scanf("%d", &b);
```

```
整数A : 10
整数B : 3
AをBで割ると3あまり1です。
```

```
if (a (44) b)
    printf("AをBで割ると%dあまり%dです。\\n", a (45) b, a (46) b);
else
    printf("AをBで割ると%dです。\\n", a (47) b);
```

```
printf("AをBで割ると%d", a (48) b);
if (a (49) b != 0)
    printf("あまり%d", a (50) b);
printf("です。\\n");
```

```
mod = a (51) b; /* modはint型の変数 */
printf("AをBで割ると%d", a (52) b);
if ((53))
    printf("あまり%d", mod);
printf("です。\\n");
```

■ 以下に示すのは、変数n1がn2の約数であるかどうかを表示するプログラム部分である。

```
if ((54))
    puts("n1はn2の約数ではありません。");
else
    puts("n1はn2の約数です。");
```

■ 以下に示すのは、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部分である。

```
if (no * (55) <= (56))
    puts("その値は13の平方根以下です。");
```

■ 以下に示すのは、double型の変数x1とx2の差に応じて、「それらの差は1.0未満です。」あるいは「それらの差は1.0以上です。」と表示するプログラム部分である。

```
if ((x1 - x2 < 1.0) (57) (x2 - x1 < 1.0))
    puts("それらの差は1.0未満です。");
else
    puts("それらの差は1.0以上です。");
```

```
diff = (58); /* diffはdouble型の変数 */
if (diff (59) 1.0)
    puts("それらの差は1.0未満です。");
else
    puts("それらの差は1.0以上です。");
```

■ 右に示す `switch` 文は、`int` 型の変数 `no` の値が

- 1 であれば と表示し、
- 2 であれば と表示し、
- 3 であれば と表示し、
- 4 であれば と表示する。

```
switch (no) {
    case 3 : printf("A");
    case 2 : printf("B"); break;
    default : printf("C"); break;
}
```

■ 以下に示すのは、変数 `c1`, `c2`, `c3` の最小値を `min` に代入するプログラム部分である。

```
min = c3;
if (min > c2) min = ;
if (min > c1) min = .
```

■ 以下に示すのは、`int` 型の変数 `no` の値が 50 以上であれば “★★★★” と、20～49 であれば “★★” と、0～19 であれば “★” と表示し、負であれば何も表示しないプログラム部分である。

```
if (no > )
    printf("★★★★");
 if ( >= )
    printf("★★");
 if ( >= )
    printf("★");
```

■ 以下に示すのは、読み込んだ二つの整数値が両方とも奇数か、片方が偶数か、両方とも偶数かを判断・表示するプログラムである。

```
# <stdio.h>

int main()
{
    int cnt; /* 偶数はいくつか */
    int na, nb;

    puts("二つの整数を入力してください。");
    printf("整数A : "); scanf("%d", &na);
    printf("整数B : "); scanf("%d", &nb);

    cnt = 0;
    if ( == 0) cnt = cnt + 1; /* 整数Aが偶数であれば… */
    if ( == 0) cnt = cnt + 1; /* 整数Bが偶数であれば… */

     (cnt) {
         0 : puts("両方とも奇数です。"); break;
         1 : puts("片方が偶数です。"); break;
         2 : puts("両方とも偶数です。"); break;
    }

     (0);
}
```