

プログラミング入門

2021/06/04 解説

高校 3 年 赤澤侑

2021/06/04

1 第 1 問

Problem 1 : 最大値

N 個の整数が与えられる。そのうちの最大値を出力せよ。

制約

$1 \leq N \leq 1000$ 各整数の絶対値は 1000 以下。

解答例は以下のとおりです。

Source Code (1): 最大値-解答例

```
1 #include <iostream>
2 using namespace std;
3
4 int main(){
5     int n, t;
6     cin >> n;
7     int ans = -10000;
8     for (int i = 0; i < n; ++i) {
9         cin >> t;
10        if (t > ans) {
11            ans = t;
12        }
13    }
14    cout << ans << endl;
15    return 0;
16 }
```

与えられる整数の絶対値は 1000 以下なので答えが-10000 になることはありませんから、これは事実上答えとしては「ありえないほど小さい」数値になっています。故に最大値を考えるときの初期値として有効です。

また、次のような別解が考えられます。

Source Code (2): 最大値-別解 1

```

1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int n, t;
6      cin >> n;
7      int ans = -10000;
8      for (int i = 0; i < n; ++i) {
9          cin >> t;
10         ans = max(ans, t);
11     }
12     cout << ans << endl;
13     return 0;
14 }

```

max という関数は 2 つの引数を取り^[1]、2 つのうち大きい方を戻り値として返します。また、小さい 2 つの引数を取り、小さい方の値を返す min という関数^[2]もあります。

さらにもう一つ別解を示しておきます。

Source Code (3): 最大値-別解 2

```

1  #include <iostream>
2  #include <algorithm>
3  using namespace std;
4
5  int main(){
6      int n;
7      cin >> n;
8      int a[n];
9      for (int i = 0; i < n; ++i) {
10         cin >> a[i];
11     }
12     sort(a, a + n);
13     cout << a[n-1] << endl;
14     return 0;
15 }

```

sort というのは並べ替えを行う関数で、並べ替える範囲を指定^[3]するとその範囲内のもの^[4]を昇順^[5]に並べ替えてくれます。並べ替えの後には、最大値は配列の一番最後にいるはずなのでそれを出力します。なお、この関数は $N \geq 10^7$ 以上では並べ替えに時間がかかりすぎてしまうため、使うことが難しいです。

[1] C++17 以降では 3 つ以上の数の最大値を返すことができるようになりました。その場合比較したい複数個の値を `max({a1, a2, ..., an})` のように中括弧でくくります。

[2] これも max 同様、3 つ以上の数の最小値を返すことができます。

[3] 生配列の場合、メモリ上の連続した場所に領域が取られるので、配列の最初のポインタ (すなわち配列変数) を第一引数に与え、要素数を加算したものを第二引数に与えることで、ポインタ演算によって範囲を指定することができます。std::vector などを使う場合は始点と終点のイテレータを渡します。

[4] 大小関係を定義できるものに限りです。ただし、大小関係はプログラマが定義することもできます。この場合については注 5 を参照してください。

[5] 比較関数を第三引数に渡すことで、昇順に限らず、並べ替えの順番 (例えば降順など) を指定することができます。通常では大小関係が定義されないものについてもこのようにすることでソートできます。

2 第2問

Problem 2 : コラッツ予想

整数 k が与えられる。 k が 1 になるまでに必要なコラッツ予想の試行回数を求めよ。

ただし、コラッツ予想とは任意の正整数 n に対して

(i) n が偶数のとき 2 で割る

(ii) n が奇数のとき 3 倍して 1 足す

を繰り返すと有限回の試行の後、全ての正整数が 1 に至るという予想である。

制約

$$1 \leq k \leq 100000$$

解答例は以下のとおりです^[6]。

Source Code (4): コラッツ予想-解答例

```
1 #include <iostream>
2 using namespace std;
3
4 int main(){
5     int k;
6     cin >> k;
7     int ans = 0;
8     for(; k != 1; ){
9         if (k % 2 == 0) {
10             k /= 2;
11         } else {
12             k = 3*k + 1;
13         }
14         ++ans;
15     }
16     cout << ans << endl;
17     return 0;
18 }
```

ループの仕様についてはすでにアップした「#1.pdf」を確認してください。(次ページに続きます)

[6] 例によって exclamation mark を半角で打てなかったのが全角で打っています。半角に直して実行してください。

また、「条件を満たす間ずっと繰り返す」を明示的に行う場合は次の while 文を使うことが多いです。

Source Code (5): コラッツ予想-別解 1

```
1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int k;
6      cin >> k;
7      int ans = 0;
8      while(k != 1){
9          if (k % 2 == 0) {
10             k /= 2;
11          } else {
12             k = 3*k + 1;
13          }
14          ++ans;
15      }
16      cout << ans << endl;
17      return 0;
18 }
```

while() の小括弧の中に条件式を書きます。この条件を満たす間ループします。評価のタイミングは for 文と同じように「ループの中身」を実行する前です。