



## 技術情報 (Technical Info Sheet)

### 採点システムの制約

解答プログラムは必ず正常終了すること（必ず 0 を返すこと）。解答プログラムを採点用入力データに対して実行した際に、時間やメモリの制限を満たして正常終了し、かつ、出力が正しい場合に、採点プログラムは、その採点用入力データの配点分の得点を与える。ソースを提出する回数に制限はないが、ソースを複数回提出した場合は最後に提出したソースのみが採点の対象となる。

### C++ の入出力について

大量の入出力データを扱う課題を解く際に、入出力の処理に `cin`, `cout` ストリームを使用した C++ プログラムは `scanf`, `printf` 関数を使用した同等のプログラムに比べて遅い。 `cin` / `cout` ストリームを使用しているのであれば、代わりに `scanf` / `printf` を使用することを強く薦める。

### 64 bit データ型

課題によっては、32 bit には収まらない大きな整数を扱う必要がある。この場合、C/C++ では `long long` などの 64 bit の整数データ型を用いないとならない。以下に、これらのデータ型の使い方を説明するためのサンプルコードを挙げる。

#### C/C++

```
int main(void) {
    long long varname;

    scanf("%lld", &varname);
    // 変数 varname に対して何か処理をする
    printf("%lld\n", varname);

    return 0;
}
```

### スタック制限

特に指定のない限り、競技システムで実行される際のスタックのサイズに制限はない。ただし、使用されるスタックは、そのプログラムが使用するメモリの一部として扱われるので、Overview Sheet に記載されたメモリ制限を超えることはできない。再帰処理を行うときはスタックオーバーフローに注意すること。