



## 技術情報 (Technical Info Sheet)

### 競技システムについて

- 解答提出用 Web インタフェースからソースを提出すること。解答プログラムは、入力を標準入力から読み込み、出力を標準出力に出力し、正常終了すること (0 を返すこと)。各課題ごとに、最後の提出から 1 分間は解答を提出することができない。ただし、課題によってはこれとは異なる解答提出方法のものもあるかもしれない。その場合、詳細は問題文および Overview Sheet に記載される。
- 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、その採点用入力データの配点分の得点が与えられる。ただし、課題によってはこれとは異なる採点方法のものもあるかもしれない。その場合、詳細は問題文および Overview Sheet に記載される。
- ソースを複数回提出した場合は、提出された全てのソースが採点され、提出されたソースの得点の最大値がその課題の得点となる。ただし、競技サーバに予期せぬ障害が発生した場合は、最後に提出されたソースのみを採点の対象とする場合がある。提出したソースは、各自の競技用 PC に保存しておくことを勧める。
- 採点システムで実行される際のスタックサイズに制限はない。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。競技用 PC 上で実行する際にスタックサイズの制限を解除するには、シェル上で `ulimit -s unlimited` を実行すること。このコマンドを実行することで、そのシェルを閉じるまでは、そのシェル上で実行されるプロセスに対してスタック制限が解除される。

### 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。

```
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;
}
```