

## 2013 年問題

すごろくというゲームがある(漢字では双六という)。盤面にマスが並んでおり、サイコロを投げて出た目の数だけマスを進んで、早くゴールについた人が勝ちという単純なゲームだ。昔はなぜかお正月のゲームの定番だったのだが、さて今の高校生にはなじみがあるのだろうかないのだろうか。これをもっと複雑にすると人生ゲームやモノポリーなどのゲームになる。すごろくではスタートのマスは「ふりだし」、ゴールは「あがり」と呼ばれる。

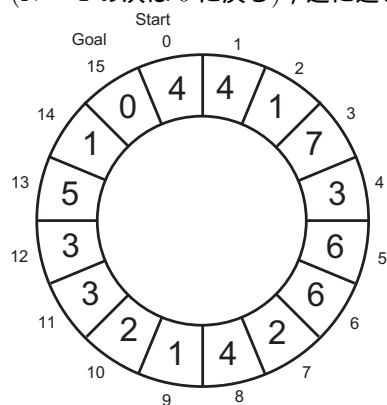
さて、すごろくのマスの中には「マス先に進む」とか「マス戻る」とか「一回休む」とかいう指示が書かれたものがある。ちょうどそのマスに止まったら、指示に従わなくてはならない。いちばん悲しいのは「ふりだしに戻る」だ。

今回は、サイコロを振らずにマス目の指示だけに従って移動するようなすごろくを考えよう

### 【問題共通の設定】

図のように円周状にマスがならべられている。マスには数字が書かれている。また、スタートとゴールのマスも決められている。最初、すごろくのコマはスタートに置かれている。そこから、マスに書かれた数字に従って進み、ゴールを目指す。マスの数字が0のときは、それ以上どこへも進めないで終了となる。

マスの数  $N = 16$  の例を図に示す。マスには0から15までの番号がつけられている(円周の外側の数字)。各マスには  $N/2 - 1$  以下の数字(この例では0から7まで)がひとつ書かれている。問題全体を通してスタートは0番でゴールは  $N - 1$  番とする。マスの番号が0,1,2と増える方向に進むことを右回り( $N - 1$ の次は0に戻る)、逆に進むことを左回りと呼ぶ



### 問A (スーパーコン 3 級問題)

スタートから右回りにマスに書かれた数字だけ進む。図の例では、スタートのマスに4と書かれているので右回りに4マス進んで、4番に着く。4番のマスには3と書かれているので、右回りに3マス進め7番に着く。これを繰り返してコマを進めていく。ゴールのマスにぴったり止まったときだけゴールでき、ゴールに止まらずに行き過ぎてしまったら、さらに同じようにコマを進め続ける。ゴールに止まるまで何周してもよい。

では、数字にしたがって何周も回っていれば、かならずいつかはゴールに到達できるのだろうか。そうではない。数字がどういう順番で書かれているかによって、いずれはゴールに到着できる場合もあれば到達できない場合もある。図の例は1周でゴールに到達できる場合になっており、スタートからゴールへの「道順」(コマが止まるマスの番号を進む順に並べたもの)は0, 4, 7, 9, 10, 12, 15である(確認してみよう)

そこで問題．スタートのマス目から右回りに各マスに書かれている数字が入力として与えられる（最後のマス目，つまり，スタートの左隣のマス目がゴール）．このすごろくで，スタートに置かれたコマがマス目に書かれた数字に従って右に進んでいくとき，いずれはゴールに到達できるかどうかを判定し，到達できる場合には，その道順を答えよ．

ヒント：ゴールに到達できないときは永久に進み続けるわけだが，マス目の総数は決まっているのだから，永久に進み続けるというのは結局，ゴールに到着する前に同じマス目に戻ってくる，ということだ．

#### 問B（スーパーコン 2 級問題）

すごろくのルールの基本は問Aと同じ．ただし，今度は左右どちらにでも進めることにする．たとえば 3 と書かれていれば，右に 3 マスまたは左に 3 マスのどちらを選んでよい．したがって，問Aと違い，各マス目に止まるたびに進み方は 2 通りあるので道順は何通りも考えられる．図では 0, 12, 15 もゴールに到達できる道順になっている

問題は問Aと同様．ただし，道順は何通りも考えられるので，そのうちでスタートからゴールに到達する道順が 1 つでもあるかどうかを判定し，到達できる場合にはその道順（のうちのどれか 1 つ）を答えよ．

#### 問C（予選問題兼スーパーコン 1 級問題）

すごろくのルールの基本は問Bと同じだが，同じマス目に 2 度以上止まってはならないことにする．

問題も問Bと同様．ただし，スタートからゴールに到達する道順の中で，（右に進むマス目数）－（左に進むマス目数）が最大となるものの道順を答えよ（複数あった場合には，そのうちのどれか 1 つ）．同じマスに 2 度以上止まってはならないことに注意．

図で可能な道順のひとつは 0, 4, 7, 5, 11, 14, 13, 2, 3, 10, 12, 15 で，これは右方向に 34 マス，左方向に 3 マス進んでいるので，（右に進むマス目数）－（左に進むマス目数）は 31 である（確認しよう）

### 解答方法 (すべての問題に共通)

A-C いずれも複数の問題が順に与えられるので、それらを制限時間以内に解くプログラムを作成して提出する。

問題のサイズと問題数, 制限時間 (全問題を解くまでの制限時間)

- 3 級 N=50 5 問 1 分
- 2 級 N=20 5 問 2 分
- 1 級 N=30 5 問 15 分

### 作成するプログラム

提出するプログラムはチーム名をファイル名とする単一ファイルとする。プログラムは C 言語規格 (ANSI C や C99 など) に準拠する C 言語で記述 (インラインアセンブラの使用は禁止)。標準 ライブラリ関数 (stdlib, mathlib 等) を宣言して使用してもよい。

プログラムの出力には用意した関数を使う (ヘッダー・ファイル `sc13.h` を使用。これについては次ページで説明)。この関数は書き換えないこと。また、それ以外には出力関数が含まれないようにすること (デバッグ用の出力などは必ず除いておく)。

`int` は 32 ビット, `long long` は 64 ビットを仮定する。long のビット幅は環境により異なるので, 64 ビットデータを扱いたい場合は `long` ではなく `long long` を使うこと (`stdint.h` をインクルードして使える `int64_t` などを使うとさらに確実)。審査は, Linux 上の `gcc ver.4` を使用してコンパイルし, 64 ビット x86 64 版の Linux 上で実行する。移植性の問題 (例えばエンディアンの違い) によるトラブルには対処しない。審査環境はクロックが 2.5GHz 程度, メモリはおよそ 4 ギガバイト。

### 審査方法 (スーパーコン認定に関して)

応募プログラムをコンパイルして実行し, 制限時間以内にすべての問題に正確な答えを出した場合に合格とし, 各級を認定する

### 審査方法 (予選選抜に関して)

応募プログラムをコンパイルして実行し, 制限時間内に正確な答えを出した問題の個数順をもとに順位を決める。同順位のチームに対しては, 正解を出した計算の計算時間の合計を用いて順位を決める (万が一, 合計計算時間にほとんど差が無い場合には, 問題数を増やすことがある。それでも差がつかないときは, 合わせて提出されるプログラムの説明をもとに工夫度を評価して順位を決める)。以上の順位付けのもとで, 東西上位 10 チームずつを本選出場候補チームとして選抜する。ただし, 1 校からは最大 2 チームしか本戦出場できない。

[注意]: スーパーコン 1 級の認定条件と予選選抜の条件は違うので, 予選で選抜されても 1 級認定にならない場合がある

入力について (すべての問題に共通)

問題ファイルには各マスに書かれた数字が 0 番のマス (スタート) から  $N - 1$  番のマス (ゴール) まで 1 行にひとつずつ、番号順に書かれている。  $N - 1$  番の次には、引き続いて次の問題の 0 番が書かれている。つまり、  $N = 50$  で問題数が 5 問のときは、250 個の数字が 1 行にひとつずつ書かれている。プログラムはそのようなデータを標準入力から読み込むように作り、実行時には問題ファイルをリダイレクトする。たとえば、`s[50]` という整数配列に標準入力から 50 個の数値を読み込むなら、いちばん簡単には

```
for(i=0; i<50; ++i)scanf("%d",&s[i]);
```

でよい。

各級ごとの問題ファイルのサンプルをひとつずつ用意してある。それぞれ、5 問分がひとつのファイルになっており、解のない例も含まれている。

出力について (すべての問題に共通)

出力用の関数を含むヘッダー・ファイル `sc13.h` を用意してあるので、かならずそれを用い、独自の出力関数を使わないこと。また、`sc13.h` は書き換ええないこと。`sc13.h` をプログラムと同じディレクトリに入れて

```
#include<stdio.h>
#include "sc13.h"
int main()
```

などとする。出力の関数はヘッダーの中で

```
void sc_output(int sc_length, int *sc_path)
```

と定義されている。解答を出力するときは、道順を `int` 型の整数配列に入れ、道順に含まれるマスの数とともにこの関数にわたす。道順にはスタートとゴールも含まれる。たとえば、プログラム中で `int` 型の整数配列 `path` を定義して (名前はなんでもよい)、その中に道順に含まれるマスの番号を訪れる順番に並べたとしよう。問 C の問題文に挙げた例なら、`path` には

0, 4, 7, 5, 11, 14, 13, 2, 3, 10, 12, 15

をこの順番に格納する。この道順に含まれるマスの数は 12 である。このとき、出力は

```
sc_output(12, path)
```

とする。第一引数が道順に含まれるマスの数である。

ゴールに到達できる道がないときには、第一引数を 0 とする (第二引数には配列を与えるが、中身はなんでもよい)。

```
sc_output(0, path)
```