ABC 126 解説

writer: tozangezan, gazelle, drafear

2019 年 5 月 19 日

A: Changing a Character

問題文のとおりに文字列の K 番目の文字を小文字に変換すればよいです。

A, B, C の 3 文字のみが入力に含まれるので if 文を 3 個書いてもよいですし、アルファベットの大文字と小文字の文字コードが 32 異なるということを利用して char 型の該当する文字に 32 を加えても良いです。

C++ での実装例は以下の通りです (include 等は省いています)。

```
char in[55];
int main(){
  int a,b;scanf("%d%d",&a,&b);
  scanf("%s",in);
  in[b-1]+=32;
  printf("%s\n",in);
}
```

B: YYMM or MMYY

まず、与えられる入力を長さ 4 の文字列とみなしたとき、前半 2 文字から 2 桁の整数と後半 2 文字から 2 桁の整数に変換してみましょう。例えば前半 2 文字から 2 桁の整数に変換するのは、 `(s[0]-'0')*10+s[1]-'0'` などと計算することで得られます。

代わりの手段として、入力が数字列であるため、これを整数とみなし、4 桁の整数から 2 桁の整数を 2 つ取り出すこともできます。この場合では、前半の 2 桁の整数は例えば `a/100` 、後半の 2 桁の整数は `a%100` となります。

その後、各種フォーマットの条件を満たすかを判定します。

YYMM フォーマットであるためには、後半の 2 桁の整数が 1 以上 12 以下である必要があります。

MMYY フォーマットであるためには、前半の 2 桁の整数が 1 以上 12 以下である必要があります。

上の 2 つを判定することで、4 つの答えのうちどれであるかが決まります。

C++ での実装例は以下の通りです。

```
int main(){
  int a;scanf("%d",&a);
  int L=a/100;
  int R=a%100;
  if(1<=L&&L<=12){
    if(1<=R&&R<=12)printf("AMBIGUOUS\n");
    else printf("MMYY\n");
  }else{
    if(1<=R&&R<=12)printf("YYMM\n");
    else printf("NA\n");
  }
}
```

C: Dice and Coin

最初に出たサイコロの目ごとに処理していき、最後にそれぞれの確率を合計します。

それぞれ、何回コインの表を出し続けることで初めて得点が K 以上になるかは、ループで数えることで求めることができます。この回数だけコインが表を出し続けることが必要です。(途中で裏が出てしまったら得点は 0 となるので、勝つことはできません)。

サイコロを 1 回振り i $(1 \leq i \leq N)$ が出る確率は $1/N$ です。そのそれぞれに対し、 t 回コインの表を出し続ける必要がある場合、成功率は 0.5^t となります。$1/N \times 0.5^t$ のすべての合計が求める答えになります。

以上の方法で計算すると、計算量は $O(NlogK)$ となり、間に合います。

C++ での実装例は以下の通りです。

```
int main(){
  int a,b;scanf("%d%d",&a,&b);
  double ret=0;
  for(int i=1;i<=a;i++){
    double tmp=1.0/a;
    int now=i;
    while(now<b){
      now*=2;
      tmp/=2;
    }
    ret+=tmp;
  }

  printf("%.12f\n",ret);
}
```

D: Even Relation

以下、与えられたグラフを適当な頂点を根とみなした根付き木として考えます。また根から頂点 i への距離を d_i とします。

任意の 2 頂点 u と v について、その最小共通祖先を w とすると、u と v の距離は $d_u + d_v - 2d_w$ と書くことができます。この式の第 3 項は偶数なので、d_u と d_w の偶奇が等しいときに限り、u と w の距離は偶数になります。よって例えば d_i が偶数の頂点は白に、奇数の頂点は黒に塗ることで条件を満たす塗り分けが可能です。

E: 1 or 2

各条件は

- $Z_i \equiv 0 \pmod 2$ のとき

$$A_{X_i} = A_{Y_i}$$

- $Z_i \equiv 1 \pmod 2$ のとき

$$A_{X_i} \neq A_{Y_i}\ (A_{X_i} = 3 - A_{Y_i})$$

です。すなわち、A_{X_i} または A_{Y_i} の一方が分かればもう一方を知ることができます。

したがって、$(X_1, Y_1), ..., (X_M, Y_M)$ 間にのみ辺が存在する N 頂点のグラフ G を考えると、i 番目のカードをめくると G 上で頂点 i と同じ連結成分の頂点に対応するカードの数字が分かります。よって、G の連結成分数が答えになります。

これは、塗りつぶしの要領で深さ優先探索を行うことや、Union-Find を用いることで $O(N+M)$ または $O((N+M)\alpha(M))$ の計算時間で求めることができます。ここで、α はアッカーマン関数の逆関数ですが、非常に小さいため、ほとんど定数と同じと考えて構いません。

おまけ (Hard バージョン)

問題 入力に矛盾がある場合は -1 を出力せよ。

解説

各条件は

$$A_{X_i} + A_{Y_i} \equiv Z_i \pmod 2$$

と表すことができます。

さらに、$i = 1, 2, ..., N, j = 1, 2$ に対して、$p_{i,j} \in \{\mathrm{true}, \mathrm{false}\}$ を

$$p_{i,j} = \mathrm{true} \Leftrightarrow i \text{ 番目のカードに } j \text{ が書かれている}$$

と定義します。すると、各条件は

- $Z_i \equiv 0 \pmod 2$ のとき

$$p_{X_i,0} = p_{Y_i,0}$$

$$p_{X_i,1} = p_{Y_i,1}$$

- $Z_i \equiv 1 \pmod 2$ のとき

$$p_{X_i,0} = p_{Y_i,1}$$

$$p_{X_i,1} = p_{Y_i,0}$$

です。このとき、$i = 1, 2, ..., N$ について

$$p_{i,0} \neq p_{i,1}$$

であることを確認すれば良いです。各 $p_{i,j}$ に対応する頂点を作った Union-Find を考えます。各 $p_{i_1,j_1} = p_{i_2,j_2}$ に対してこれらに対応する頂点同士を unite します。最後に、$p_{i,0} \neq p_{i,1}$ に対して、これらに対応する頂点同士が unite されていない (同じグループに属していない) かを調べることで矛盾があるか確認することができます。

F: XOR Matching

$K \geq 2^M$ のとき、明らかに解は存在しません。なぜなら 2^M 未満の整数同士の排他的論理和をとっても 2^M 以上になることはないからです。以下、そうでないときを考えます。

$M = 0$ のとき、a は一意であり $K = 0$ の解です。

$M = 1$ のとき、$K = 0$ では解が存在し $K = 1$ では存在しないことが全列挙などで分かります。

$M \geq 2$ のとき、0 以上 2^M 未満の任意の K について解が存在します。このことを具体的に a を構成することで示します。

K 以外の 0 以上 2^M 未満の整数を昇順に並べた数列を b とします。また b の逆順を c とします。このとき b K c K と並べてできる数列を a とすればよいです。

というのも、まず任意の i $(0 \leq i < 2^M, i \neq K)$ について、この数列中の 2 つの i の間には、K が 1 回、それ以外の数が 2 回現れます。よって K 以外の数は打ち消し合って、題意の排他的論理和は K になります。

次に、この数列中の 2 つの K の間には、K 以外の 0 以上 2^M 未満の整数が 1 回ずつ現れます。これらの排他的論理和は、演算の性質を考えると $(0 \oplus 1 \oplus ... \oplus (2^M - 1)) \oplus K$ と書けます（$\oplus$ は排他的論理和演算を表します)。この式の括弧で囲った部分については、排他的論理和が $2^M - 1$ になるペアを偶数個作れるので 0 になります。つまり式の値は K になります。

よって K に関しても題意の排他的論理和は K であり、この数列は与えられた条件を満たすことが分かります。