

第四回全統かんだ模試

ボイヤー・ムーア法とは、文字列検索に用いられるアルゴリズムである。ここでいう文字列検索とは、ある長さを持つテキストについて、その長さ以下のパターンが部分文字列として含まれているかを検索することである。あらかじめパターンについて前処理を行っておき、その情報を用いて、文字の不一致が生じた場合に検索文字をスキップすることで処理を削減する。具体例を以下に示す。

例) テキスト `abebadcb` から `adc` を検索する

パターンに現れる `a,d,c` についてスキップ表を作成する。

これは、パターンの末尾からの文字数を示す。

パターンに登場しない文字に関しては、パターンの長さとする。

テキストの注目している文字とパターンの間に不一致が生じたとき、テキストで注目している文字ごとに注目する文字をスキップする。

A	d	C	それ以外の文字
2	1	0	3

文字列の比較の際には、パターンの末尾から照合する。

テキスト 3 文字目の `e` とパターン 3 文字目の `c` を照合する。

不一致しているので、テキスト 3 文字目の `e` とスキップ表を照合する。

`e` を読み込んだ際には 3 文字スキップするので、テキストの 6 文字目に注目する。

テキスト 6 文字目の `d` とパターン 3 文字目の `c` を照合する。

不一致しているので、テキスト 6 文字目の `d` とスキップ表を照合する。

`d` を読み込んだ際には 1 文字スキップするので、テキストの 7 文字目に注目する。

テキスト 7 文字目の `c` とパターン 3 文字目の `c` を照合する。

一致しているので、1 文字ずつ前に注目する。

テキスト 6 文字目の `d` とパターン 2 文字目の `d` を照合する。

一致しているので、1 文字ずつ前に注目する。

テキスト 5 文字目の `a` とパターン 1 文字目の `a` を照合する。

一致しており、パターンを全て照合したので検索成功。

このアルゴリズムを、0~9の整数を用いた数列について用いることを考える。次のプログラムはC言語を用いた、長さMAXの数列numから、パターン数列ptnを関数searchを用いて検索するプログラムである。検索に成功すれば、numにおけるptnの先頭のインデックスを出力し、失敗すればその旨を出力する。このプログラムについて以下の間について答えよ。

1. 29行目で宣言されているptnの内容が以下の場合における出力を示せ。
 - i. {4,3,7,0,2}
 - ii. {0,2,4,3,0}
2. 12行目において、パターン中の各文字についてスキップ数を決定している。このとき、パターン中に同じ文字が含まれていた場合、その文字についてのスキップ数はどのように決定されるか答えよ。
3. 空欄（ア）～（ウ）を適切に埋めよ。
4. 22~23行目において、図のような記述の場合、パターンによっては無限ループが発生してしまう場合がある。
 - i. 無限ループが発生してしまうようなパターンを以下の中から1つ選べ。またその理由を簡潔に述べよ。
 - ① {3,4,3,4,4}
 - ② {1,8,0,2,4}
 - ③ {4,3,7,2,4}
 - ④ {1,9,5,7,9}
 - ii. このような無限ループが発生しないように、22~23行目の枠内を変更する。変更後を以下のようにするとき、空欄（エ）を埋めよ。

```
22     if( (エ) < j)    i+=skip[txt[i]];
23     else            i+=PTN-j;
```

```

1  #include <stdio.h>
2  #define PTN 5
3  #define NUM 15
4  #define MAX 10
5  int search(int ptn[],int txt[]){
6      int i,j;
7      int skip[MAX];
8      for(i=0;i<MAX;i++){
9          skip[i]=PTN;
10     }
11     for(i=0;i<PTN;i++){
12         skip[ptn[i]]=PTN-i-1;
13     }
14     i=PTN-1;
15     while(i<NUM){
16         (ア)
17         while(ptn[j]==txt[i]){
18             if(j==0) return i;
19             (イ)
20             (ウ)
21         }
22         i+=skip[txt[i]];
23     }
24     return -1;
25 }
26
27 void main(){
28     int num[NUM]={5,7,9,8,2,0,2,4,3,7,0,2,4,1,9};
29     int ptn[PTN]={4,3,7,0,2};
30     int ans=search(ptn,num);
31     if(ans!=-1) printf("index:%2d¥n",ans);
32     else printf("Not found¥n");
33 }

```