

数値解析

第4回

2023年10月26日

データ構造と組み合わせ評価の最適化

(順列) の準備・・・配列

figure[i][j]

<div><div>j</div><div>i</div></div>	0	1
0	3	—
1	0	8
2	1	5
3	2	3

- 整数型の2次元配列figure[100][2]を用意する。
- figure[i][1]に数値を入れる。
- figure[0][1]は空けておく。ここで、一番小さい数値の入っている配列の添字iをfigure[0][0]に入れる。
- figure[i][0]に次に小さい数値の入っている位置の添字を・・・と繰り返すと左のようになる。
(この様な連なりの事をチェインと呼ぶ)

順列のプログラミング

【プログラム全体像】

このプログラムを実現するため、0を入力するまで、1～100の値をキーボードから100個以内入力し、`figure[i][1]`に順に代入する。この時、`figure[i][0]`には値0を代入しておく。

【プログラムの各処理】

1. キーボードから最初に値を入力する時に、まず`figure[1][1]`に値を代入し、`figure[1][0]`に0を入れる。また、`figure[0][0]`に最初の値を入れた位置(つまり1)を代入する。
2. 2番目以降の値を入力するために、for文で*i*=2～99まで以下を繰り返す。
 - 2.1. 値を入力し、もし値が0なら3.に行く。
 - 2.2. 入力した値を`figure[i][1]`に代入し、`figure[i][0]`は0を代入する。次に`figure[0][0]`の値を変数pointに代入し、変数preに0を代入する。
 - 2.3. `figure[point][1]`の値が`figure[i][1]`の値よりも小さいならpreにpointの値を代入し、pointに`figure[point][0]`の値を代入する。逆に大きい又は等しいなら`figure[pre][0]`にiを代入し、`figure[i][0]`にpointの値を代入後に2.1へ行く。
 - 2.4. pointの値が0であれば`figure[pre][0]`にiを代入し2.1へ、0で無ければ2.3へ行く。
3. pointに`figure[0][0]`を代入する。
4. pointが0になるまで`figure[point][1]`の値を出力し、pointに`figure[point][0]`の値を代入する。
5. 終了。

以上の結果、数値が小さい順に並んで出力されることを確かめる。

以上のプログラムが出来た人は

- プログラムの最後で最小値と最大値を出力するプログラムを作成する。
- 以上のプログラムで出来た昇順（小さい順）に並んだチェインを降順（大きい順）に並べ換えるプログラムを作成する。
- 2次元配列を`figure[100][3]`とし，大きい順にも小さい順にもチェインをたどれる様に両方向にチェインを作るプログラムを作成する。

ヒントのプログラム例（一部略）

```
#include <stdio.h>
#include <math.h>
int main(void){
    [略]
    printf("figure[1][1]="); scanf("%d",&figure[1][1]);
    figure[1][0]=0; figure[0][0]=1; pre=1; point=1; /* 1. 最初の値代入 */

    for(i=2;i<100;i++){
        printf("figure[1][%d]=",i); scanf("%d",&figure[i][1]); /* 2. 二番目以降の値入力 */
        if(100<figure[i][1]){
            printf("Input error¥n");
            i--; continue;
        }
        if(figure[i][1]==0){ break;} /* 2.1. もし0が入力されたら終了 */
        figure[i][0]=0; point=figure[0][0]; pre=0; /* 2.2. 次のループ用の前準備 */
        while(1){ /* 新しいデータをチェインのどこに入れるか探す */
            /* 2.3. チェインの間の入れるべき位置を探す */
            if(figure[point][1]<figure[i][1]){ pre=point; point=figure[point][0];}
            else{ figure[pre][0]=i; figure[i][0]=point; break;}
            /* 2.4. 最後まで探し終わったので最後尾に */
            if(point==0){ figure[pre][0]=i; break;}
        }
    }
    /* 3. pointが0になるまでfigure[point][1]の値を出力 */
    point=figure[0][0];
    while(point!=0){
        printf("figure[%3d][1]=%3d¥n",point,figure[point][1]);
        point=figure[point][0];
    }
    [略]
}
```

課題

チェーンおよび線形リストについての説明を,
Moodle上から回答せよ.

【再通知】 レポート課題1

Product mix問題を解くC言語のプログラムのソースファイルをMoodle上から期限厳守にて提出せよ。

提出期限：**2023年11月2日(木)23:59:59【厳守】**

- レポート課題の不備や未完了は**減点・再提出**
(他者のレポートコピー等**不正行為厳禁**)