

メディアプログラミング演習—第9回（第4テーマ2日目）—

第4テーマ：領域埋めアルゴリズム（その2）

平面上の複雑な形をした領域を「塗りつぶす」アルゴリズムとしてスキャンライン法を扱った。本演習で言う「領域」とは、前回同様に「画像の中心を含む、黒色のピクセルが繋がってできている閉領域」を想定する。2日目では、シードフィル (seed fill) 法による領域埋めを扱う。

演習9：シードフィルによる塗りつぶし

(1) 概要

シード・フィル(seed fill)による閉領域の塗り潰しアルゴリズムは、以下の流れである。

1. 塗り潰しの開始点（シード）として指定した 1 ピクセルを塗る。
2. 今塗ったシードピクセルの周囲から塗り潰すべきピクセルを探し、シードの候補としてその座標をスタックに記憶する。
3. スタックから 1 点取り出し、そのピクセルが白ピクセルなら、それをシードとして 1,2 の処理を繰り返す。
4. スタックが空になった時点で塗り潰し処理は完了する

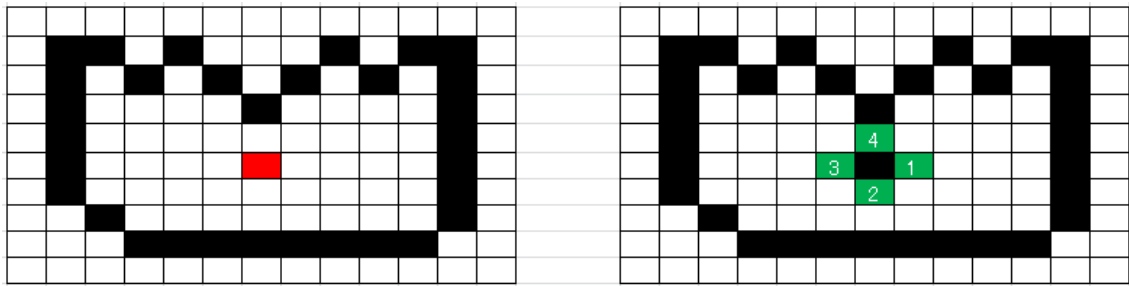
上記2のステップにおける「塗り潰すべきピクセル」として、その時点のシードの上下左右の4ピクセルをその候補とする。さらに詳細化すると以下となる

0：画面中央の座標 ($\text{width}/2, \text{height}/2$) をスタックにプッシュする。

1：スタックが空でない間、以下を繰り返す。

- 1.1:スタックからポップする。それを (x,y) とする。
- 1.2:ピクセル (x,y) が白なら、以下 1.3 から 2.4 を行う。
 - 1.3：そのピクセルを黒にする。
 - 2.1：ピクセル $(x+1,y)$ が白なら、 $(x+1,y)$ をプッシュする。
 - 2.2：ピクセル $(x,y+1)$ が白なら、 $(x,y+1)$ をプッシュする。
 - 2.3：ピクセル $(x-1,y)$ が白なら、 $(x-1,y)$ をプッシュする。
 - 2.4：ピクセル $(x,y-1)$ が白なら、 $(x,y-1)$ をプッシュする。

上記アルゴリズムにおいて、最初にループのステップ 1.2 の時点は下図左であり、シードとなるピクセルを赤で示す。



(a) 最初のループのシード

(b)最初のループでのスタック状況

図9-1 シードフィルアルゴリズムにおける最初のループの状況

ステップ 1.3 から 2.4 まで行くと、上図右となる。緑セルがシード候補としてスタックされる。番号はスタックのポインター値である（便宜上、ポインターは1からとする）。2回目のループにおける状況を以下に示す

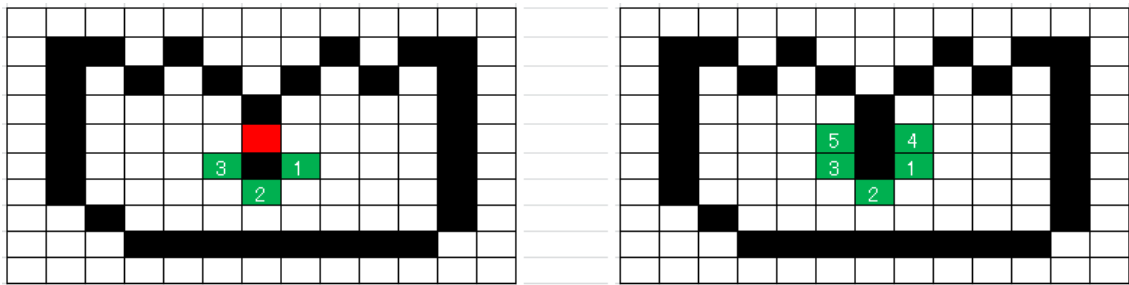


図9-2 シードフィルアルゴリズムにおける2巡目のループの状況

(2) プログラムの概要

処理プログラムの概要は以下の通りである

```
<<大域変数として、x,y を宣言しておく>>
ClearStack(); PushStack(width/2,height/2);
While(StackPointer>0){
    PopStack( );
    If( Is_White(x,y)==1){
        Set_Black(x,y);
        << この部分に 2.1 から 2.4 の処理 >>
    }
}
```

(3) 演習

配布プログラム `sample4-2.txt` を参考にシードフィル法による領域埋めプログラムを完成させなさい。