

条件分岐 (つづき)

■ 無限ループと if, break

サンプルプログラム para1.c をまず実行して出力結果を確認せよ。白い□が右に描かれるはず。

```
int main() {
    int win;
    double x,y,d,w;

    win=gopen(400,400); /* 描画ウィンドウを開く */
    winname(win, "para"); /* 名前をつける */

    x=0.0; y=0.0; /* 初期位置を設定 */
    w=10.0; /* 四角の幅を設定 */
    d=w*1.5; /* 位置の移動は四角の幅の 1.5 倍 */

    newpen(win, 1); /* 色を白に設定 */
    while(1) {
        if((x+w)>400) { /* 描こうとする四角の右端がはみ出たら */
            break; /* 処理終了 */
        };
        fillrect(win, x, y, w, w);
        x+=d; /* ひとつ右に移動 */
        usleep(50000);
    };

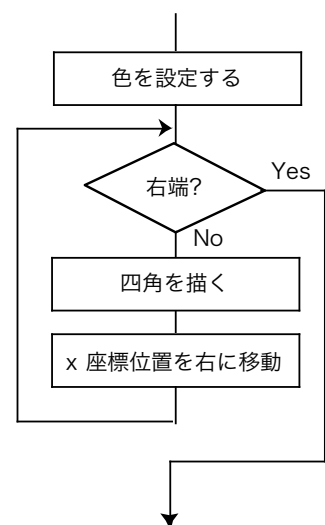
    ggetch(win);
    gclose(win);
    return(0);
}
```

このプログラムの考え方：

1. x,y という変数で座標位置を覚えておき、
1. それを少しずつ右にずらして□を描き、
1. 右端に到達したら終了する。

今回のポイントは：

- while() によるループ (教科書 pp.130) があるが、それは条件節に 1 と書かれた無条件ループ (無限ループ) である。(教科書 pp.133)
- 右端に到達したかどうかをif文 (教科書 pp.112) によって毎回確認し、ループを終了 (脱出) させる。
- while() ループからの脱出には break を使う。(教科書 pp.144)
- usleep() によって 50ms 待ち時間が入っている。(そうしないと速すぎて面白くない)



■ 課題 1.

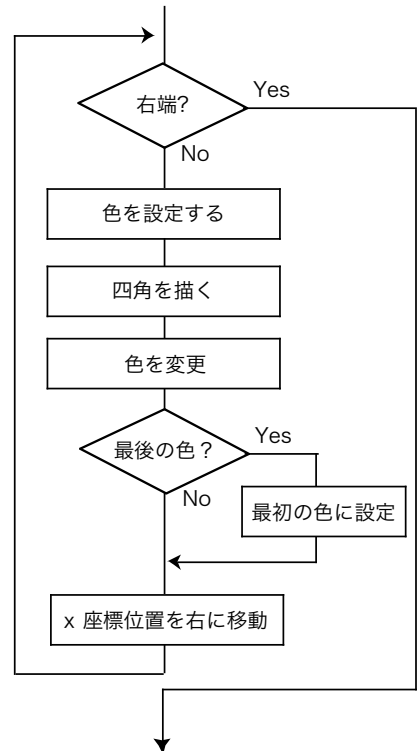
サンプルプログラム para1.c を元に、一つずつ四角の色を変えよ。

考え方。

1. 色番号を保持するための変数（例えば int 型で c という名前にする）を一つ用意し、
2. 毎回描くたびに色を変えていく。0 が黒なので 1 から 15 まで毎回 1 ずつ加えていくのが簡単。
3. ただし、16 という色はないので、15 まで加えたら、その次は 1 になるように条件判断をする。

ポイント：

- 色を設定しているのは
`newpen(win, 1); /* 色を白に設定 */`
の部分。第二引数が色番号になる。0 が黒なので、0 は使わないように。（背景色と同じなので、描いたものが見えない）
- para1.c では色の設定はループの外だったが、ループごとに色を変えたいのだから、`newpen()` 関数はループの中に入れる必要がある。



できあがれば講師に見せて OK を貰ったものを課題提出システムで登録する。

■ 課題 2.

課題 1. ができたら、今度は「右端まで到達したら一段上の列に口を描き続ける」ようにせよ。

考え方。

1. 右端に到達したら、y 座標を少し上に移動させ、x 座標を 0（左端）に設定する。
2. ただし、y 座標がもう一番上に到達していたら、そこで終了。

ポイント：

- if 文による条件判断が二重になる。If 文の中に if 文を入れるのは、例えば

```
if( 右端? ){  
    if( 上端? ){  
        .....
```


};
};というようにすれば良い。

できあがれば講師に見せて OK を貰ったものを課題提出システムで登録する。

