

基礎プログラミングおよび演習 レポート # 15

1710021, 有馬海人 (ペア: 鈴木順仁・1710333, 成田悠希・1710465)

02/11

1 構想・計画・設計

衝突と発散するエネルギーをアニメにしようと構想しました。
鈴木が2つの粒子の衝突、成田が衝突した後の発散、そして有馬が二人の作成した main 関数を結合させるプログラムを作成するという計画・設計であった。

2 プログラムコード

次のプログラムは `suzuki.c` という名前のものです。

```
//#include "img.h"

int suzuki(void) {
    struct color c1 = { 0, 0, 255 };
    struct color c2 = { 255, 0, 0 };
    //    struct color c3={ (c1.r+c2.r)/2, (c1.g+c2.g)/2, (c1.b+c2.b)/2 };
    struct color c3={55,255,175};
    for(int i = 0; i <=150; ++i) {
        img_clear();
        img_fillcircle(c1, i, 100, 10);
        img_fillcircle(c2, 300-i, 100, 10);
        img_write();
    }
    for(int i=0;i<=20;i=i+1){
        img_clear();
        img_fillcircle(c3, 150, 100, 10);
        img_write();
    }
    return 0;
}
```

```
}
```

次のプログラムは narita2.c というプログラムです。

```
//#include "../img.c"

int narita(void) {
    struct color c = {55,255,175};
    for(int i = 0; i <= 20; i++) {
        img_clear();
        img_fillline(c,150+10*i,100,160+10*i,100,10);
        img_fillline(c,150-10*i,100,140-10*i,100,10);
        img_fillline(c,150,100+10*i,150,110+10*i,10);
        img_fillline(c,150,100-10*i,150,90-10*i,10);
        img_fillline(c,150-10*i,100-10*i,140-10*i,90-10*i,10);
        img_fillline(c,150+10*i,100-10*i,160+10*i,90-10*i,10);
        img_fillline(c,150+10*i,100+10*i,160+10*i,110+10*i,10);
        img_fillline(c,150-10*i,100+10*i,140-10*i,110+10*i,10);
        img_write();
    }
    return 0;
}
```

次のプログラムは animate1.c という名前のものです。

```
// animate1 --- create animation using img lib.
#include "img.c"
#include "suzuki.c"
#include "narita2.c"
//#include <math.h>
//#include "img.h"

int narita(void);
int suzuki(void);

int main(void) {
    // struct color c1 = { 30, 255, 0 };
    // struct color c2 = { 255, 0, 0 };
    // for(int i = 0; i < 20; ++i) {
    //     img_clear(); img_fillcircle(c1, 20+i*8, 100, 20); img_write();
    // }
```

```

// for(int i = 0; i < 20; ++i) {
//   img_clear(); img_fillcircle(c2, 180, 100+i*5, 20-i); img_write();
// }
suzuki();
narita();

}

```

3 プログラムの説明

3.1 suzuki.c について

このプログラムは赤色の円と青色の円が合体し、その直後に蛍光緑になるというプログラムです。

3.2 narita2.c について

このプログラムは先ほどの合体していたところから緑色に八つに爆散している状況を作るプログラムです。

3.3 animate1.c について

このプログラムでは2,3,4行目ではこのプログラムを構成するにあたって重要なファイルを include しています。特に2,3行目では私以外の二人に作成してもらった関数の入ったファイルを include しています。

7,8行目では私以外の二人の作成した、関数を宣言しています。

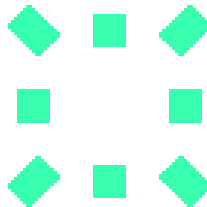
11行目ではこのプログラムにおける main 関数を定義しています。

20,21行目では二人の作成した、2つの関数を実行しています。

4 生成された動画



二つの円が合体しようとしているところ



合体した後に散り散りになっているところ

5 考察

今回の課題を行なってわかったことはチーム開発において作成したプログラムを共有する際にはどのようなプログラムを作成しているかのコメントが重要であると考察する。その理由はこのレポートを書く際にどのようなプログラムをチー

ムメンバーが書いたのかを計画段階でしか聞いていなかったため、チームメンバーの作成したプログラムを頭の中で考える必要があったからである。

次にわかったことはチーム開発の際にはまとめ役の人が必要であるということであると考察する。なぜならば今回のようなチーム開発の際に誰がどこを担当しているのかをしっかりと把握してまとめ上げる人物がいないとそれぞれが自由にコーディングしている状態である状態も考えることができる。これをこれをまとめる人がいるとそういったこともなく全体を俯瞰した立場の人が生まれるのでそれぞれがやりやすいのではないかと考察する。

最後に C 言語をチームで開発する際には個人的には、他の言語では行なったことのない、外部ファイルに存在している関数を呼び出すことができるという機能を C 言語で利用した。これは二人の作成した、main 関数となっているものを main の部分を書き換え、その入っているところを include することで関数宣言をするだけで使えるようになるからであると考察する。

以上より、この課題の目標である、「チームでソフトウェアを開発する際に注意すべきことを知る。」「C 言語の機能を活用して分担してプログラムを開発する。」は達成することができたのではないかと考察する。

6 アンケート

6.1 Q1：うまく分担して課題プログラムを開発できましたか。

分担はうまくいきました。

6.2 Q2:複数で分担する際に注意すべきことは何だと思いましたか。

それぞれがどのようなコードを書いているかの共有だと思います。

6.3 Q3：リフレクション(今回の課題で分かったこと)・感想・要望をどうぞ。

定期テストが並び替えらしく、苦手なので過去の復習問題をやり直すことで対策をしていきたいとおもいます。