

プログラミング演習I第1回演習レポート

学籍番号：2364902

名前：キム ギュソク

(1) 課題番号と課題内容

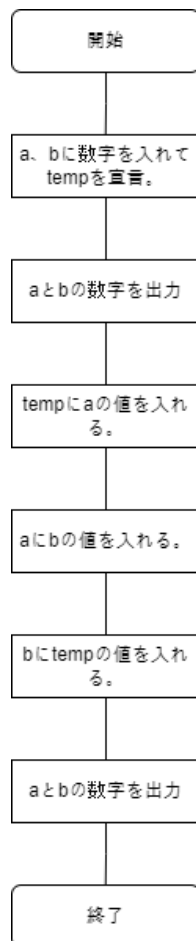
A. 課題番号：演習問題 2

B. 課題内容

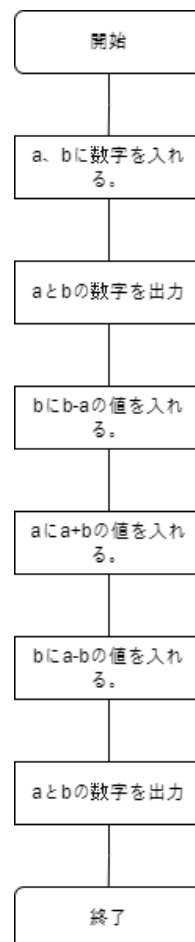
条件 I：2つの変数を交換する際に、作業用の変数を使う。

条件 II：2つの変数を交換する際に、作業用の変数を使わないで（見かけ上）交換を行う。

(2) フローチャートまたは疑似言語によるアルゴリズムの記述



条件 I



条件 II

(3) アルゴリズムが正しいことの説明

A. 条件 I

- i. aの変数に入っている数字をxと仮定してbの変数に入っている数字をyと仮定する。
- ii. temp変数にaに入っている数字を入れたのでtempにはxが入る。
- iii. a変数にb変数に入っている数字を入れたのでa変数にはyが入る。
- iv. b変数にtemp変数に入っている数字を入れたのでb変数にはxが入る。
- v. よって、a変数はy、b変数はxとなるので正しく入れ替えたのが分かる。

B. 条件 II

- i. 条件 I と同じくaはx、bはyと仮定する。
- ii. b変数にb-aを入れるとbはy-xとなる。
- iii. a変数にa+bを入れるとbはy-xなので、aはx+y-xとなりyとなる。
- iv. b変数にa-bを入れるとy-(y-x)はxとなる。
- v. よって、a変数はy、b変数はxとなるので正しく入れ替えたのが分かる。

(4) ソース・プログラムの説明

A. 条件 I

```
#include <stdio.h> //stdio.hライブラリに接続
int main(void){
    int a = 3, b = 7, temp; //a、b変数宣言と同時に3、7を入れて
    臨時の変数tempを宣言。
    printf("a = %d b = %d\n", a, b); //入れ替える前のa、bの出力
    temp = a; //臨時変数にaを入れる
    a = b; //a変数にbの値を入れる。
    b = temp; //予め入れたaに入っていた値をbに入れる
    printf("a = %d b = %d\n", a, b); //入れ替えた後のa、bの
    出力。
    return 0;
}
```

B. 条件Ⅱ

```
#include <stdio.h> //stdio.hライブラリに接続
int main(void){
    int a = 3, b = 7; //a、b 変数宣言と同時に3、7を入れる
    printf("a = %d b = %d\n", a, b); //入れ替える前のa、bの出力
    b = b-a; // a : a, b : b-a;
    a = a+b; // a : a + b - a = b, b : b-a;
    b = a-b; // a : a + b - a = b, b : b - b + a = a
    printf("a = %d b = %d\n", a, b); //入れ替えた後のa、bの出力。
    return 0;
}
```

(5) 考察

- A. プログラミングでは単純に入れ替えようと思ってa変数にbを入れてb変数にaを入れるとa変数にbを入れる時にaに入っていたデータが消えてしまう。それを防ぐために臨時の変数tempを利用するアルゴリズムが条件Ⅰである。それに比べて、条件Ⅱでは数学の原理を利用してaとbの値を入れ替えることができるアルゴリズムである。b変数にただa変数のデータを入れるのではなく、aとbデータ両方を入れることで両方のデータを守りながら入れ替えることができる。条件Ⅱは条件Ⅰに比べて宣言する変数の量が少ないのでプログラムとしては効率的だと考えられる。変数を宣言することはメモリ空間を使うことである。よって、変数を少なくして宣言する条件Ⅱのプログラムがメモリ空間が余裕があり効率的だと思う。また、コードが大きくなるとこの小さな違いが大きく影響を及ぼすので条件Ⅱの方法を利用してコードを作成する方がいいと思う。

(6) 感想

- A. 春学期に作ったプログラムをアルゴリズム的な方法で考えるいい経験でした。その時には単純に入れ替えることだけを考えたがその問題を解決するための過程について詳しく考えるようになるいい機会でした。