

知能情報工学演習I 第11回(後半第5回)

岩村雅一

masa@cs.osakafu-u.ac.jp

前回の課題

■ 課題1

- キーボードから自然数を入力し、その自然数が偶数か奇数かを判別して表示せよ。
- ヒント: 偶数か奇数かは2で割り切れるかどうかで判定可能。後半2回目で紹介したものを使用するとうまくいきます。

```
#include<stdio.h>
```

```
int main(void){
```

```
    int a;
```

```
    printf("数字を入力してください。¥n");
```

```
    scanf("%d",&a);
```

```
    if(a%2==1){
```

```
        printf("数字は奇数です。¥n");
```

```
    }else{
```

```
        printf("数字は偶数です。¥n");
```

```
    }
```

```
    return 0;
```

```
}
```

前回の課題

課題2

- キーボードから2つの整数を入力し、aとbとする(ただし、 $a < b$)。このとき、 $a + (a+1) + \dots + (b-1) + b$ を求めよ。
- ヒント: for文を使い、aからbまで足し算を実行するとよい。

レポートにあった例

- `for (a==a; a<=b; a++)`
- `for (a; a<=b; a++)`

```
#include<stdio.h>
```

```
int main(void){  
    int a,b,i,sum;
```

```
    printf("Input a: ");  
    scanf("%d",&a);  
    printf("Input b (b>a):");  
    scanf("%d",&b);
```

```
    sum=0;  
    for(i=a;i<=b;i++){  
        sum += i;  
    }
```

```
    printf("sum = %d¥n",sum);
```

```
    return 0;  
}
```

前回の課題

■ 課題3

- キーボードから5つの整数を入力し、変数(配列)で覚えておく。5つの整数の入力が終わった後、5つの値の合計、平均を求めなさい。
- ただし、平均の値は小数で出るものとする。

前回の課題：課題3

■ 配列を使わない場合

```
#include<stdio.h>
```

```
int main(void){  
    int a,b,c,d,e,sum;  
    float average;
```

```
printf("Input a: ");  
scanf("%d",&a);  
printf("Input b: ");  
scanf("%d",&b);  
printf("Input c: ");  
scanf("%d",&c);
```

```
printf("Input d: ");  
scanf("%d",&d);  
printf("Input e: ");  
scanf("%d",&e);
```

合計の計算

```
sum=a+b+c+d+e;  
printf("sum = %d\n",sum);
```

```
average=sum/5.0;  
printf("average = %f\n",average);
```

平均の計算

```
return 0;  
}
```

前回の課題：課題3

■ 配列を使う場合

```
#include <stdio.h>

int main(void) {
    int i, a[5], sum;
    float average;

    for (i=0; i<5; i++) {
        printf("Input a[%d]: ", i);
        scanf("%d", &a[i]);
    }
```

入力

合計の計算

```
sum = 0;
for (i=0; i<5; i++) {
    sum+=a[i];
}
printf("sum = %d¥n" ,sum);
```

```
average=sum/5.0;
printf("average = %f¥n",average);
```

平均の計算

```
return 0;
}
```

前回の課題

■ 課題4 (任意)

- キーボードから整数を入力してもらい、その整数が素数かどうかを判定するプログラムを作成しなさい。

[ポイント]

- 素数かどうかは、順番に割ってみればわかる
- 1は素数ではない

前回の課題：課題4

2からaまでで割る



```
#include<stdio.h>
```

```
int main(void){
```

```
    int a,i;
```

```
    printf("Input a:");
```

```
    scanf("%d",&a);
```

```
    if(a==1){
```

```
        printf("1は素数でない。¥n");
```

```
        return 0;
```

```
    }
```

1の場合

```
for(i=2;i<a;i++){
```

```
    if(a%i==0){
```

```
        printf("%dは素数でない。¥n",a);
```

```
        return 0;
```

```
    }
```

```
}
```

```
printf("%dは素数である。¥n",a);
```

```
return 0;
```

```
}
```

2以上の場合

後半の予定

7. 6月2日 プログラミング環境(テキスト1,2章)
8. 6月9日 変数とデータ型(3章)、演算子(4章)
9. 6月16日 コンソール入出力(6章)、配列(3章)
10. 6月23日 制御文1(テキスト5章)
11. 6月30日 制御文2(テキスト5章)
12. 7月14日 関数1(テキスト7章)
13. 7月21日 関数2(テキスト7章)

本日のメニュー

■ 条件の判定

- 真と偽
- 論理演算子

■ 制御文

- 繰り返し
 - while文
 - do-while文
- ループの終了
 - break
 - continue

if文

- 条件を満たすときに命令を実行する

```
int a;  
a=10;  
if (a==10) ←  
    printf("aは10です。¥n ");
```

条件を判定する仕組み

- 条件の書き方 (関係・等価演算子、テキストP.78)

- | | | | |
|------|-------|------|--------------|
| ■ == | 等しい | ■ > | 大なり |
| ■ != | 等しくない | ■ < | 小なり |
| | | ■ >= | 等号を含む大なり () |
| | | ■ <= | 等号を含む小なり () |

条件の判定

- C言語での真と偽 (条件を満たすかどうか)
 - 偽: 0
 - 真: 0以外
- 例: `a==10`の判定
 - `a`が10の場合:
 - `a==10` → 1
 - `a`が10でない場合:
 - `a==10` → 0

条件の判定のサンプルプログラム

```
#include<stdio.h>

int main(void){
    if (1) {
        printf("1は真\n");
    } else {
        printf("1は偽\n");
    }

    if (0) {
        printf("0は真\n");
    } else {
        printf("0は偽\n");
    }
    return 0;
}
```

```
#include<stdio.h>

int main(void){
    int a;
    printf("Input a: ");
    scanf("%d", &a);
    printf("a==10の値は%d\n", a==10);
    return 0;
}
```

論理演算子 (テキストP.79)

- 真偽値を否定したり、複数の条件を組み合わせる
 - NOT演算子(!)
 - If (!(a==10))
 - AND演算子(&&)
 - If (a==10 && b==20)
 - OR演算子(||)
 - If (a==10 || b==20)

論理演算子のサンプルプログラム

```
#include<stdio.h>

int main(void){
    int i;
    for (i=10; i<=20; i++) {
        if ((i%2==0) || (i%3==0))
            printf("%dは2の倍数か3の倍数である。¥n",i);
        if ((i%2==0) && (i%5==0))
            printf("%dは2の倍数でも5の倍数でもある。¥n",i);
        if ((i%2==0) && !(i%4==0))
            printf("%dは2の倍数であるが4の倍数ではない。¥n",i);
        printf("¥n");
    }
    return 0;
}
```

for文

課題2であった初期化しない例

- for (i==i; i<3; i++)
- for (i; i<3; i++)

- 条件を満たす間繰り返す

なくてもいい

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int i;
```

```
    for (i=0; i<3; i++) {
```

```
        printf("i=%d\n", i);
```

```
    }
```

```
    return 0;
```

```
}
```

初期化式

条件式

再設定式(増分)

for文

■ 初期化式が無い場合

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int i;
```

```
    i=0;
```

```
    for (; i<3; i++) {
```

```
        printf("i=%d\n", i);
```

```
    }
```

```
    return 0;
```

```
}
```

条件式

再設定式(増分)

for文

■ 再設定式(増分)も無い場合

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int i;
```

```
    i=0;
```

```
    for (; i<3; ) {
```

```
        printf("i=%d¥n", i);
```

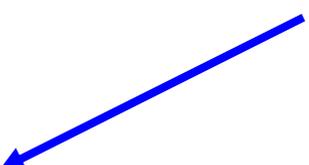
```
        i++;
```

```
    }
```

```
    return 0;
```

```
}
```

条件式



while文

□ 条件式のみ

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int i;
```

```
    i=0;
```

```
    while (i<3) {
```

```
        printf("i=%d¥n", i);
```

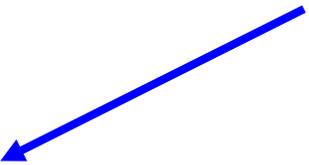
```
        i++;
```

```
    }
```

```
    return 0;
```

```
}
```

条件式



do-while文

- 基本的にwhile文と同じ
- 条件判定は後なので、最低1回は実行される

```
#include <stdio.h>
```

```
int main(void) {
```

```
    int i;
```

```
    i=0;
```

```
    do {
```

```
        printf("i=%d\n", i);
```

```
        i++;
```

```
    } while(i<3);
```

```
    return 0;
```

```
}
```

セミicolon(;)が必要

条件式

for文の無限ループ

```
#include<stdio.h>
```

```
int main(void){
```

```
  for(;;) {
```

```
    printf("a\n");
```

```
  }
```

```
  return 0;
```

```
}
```

条件判定式がない

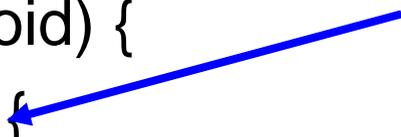


while文の無限ループ

```
#include <stdio.h>
```

```
int main(void) {  
    while (1) {  
        printf("a¥n");  
    }  
  
    return 0;  
}
```

ずっと真



ループを抜け出す方法

■ break

- 先週switch文で出てきた
- 一番内側のループを抜ける

■ continue

- 処理を1回だけスキップする

ループを抜け出す方法のサンプルプログラム

```
#include<stdio.h>
```

```
int main(void){
```

```
    int i;
```

```
    printf("break文の場合¥n");
```

```
    for(i=0; i<10; i++) {
```

```
        if (i==5) {
```

```
            break;
```

```
        }
```

```
        printf("i = %d¥n", i);
```

```
    }
```

```
    printf("continue文の場合¥n");
```

```
    for(i=0; i<10; i++) {
```

```
        if (i==5) {
```

```
            continue;
```

```
        }
```

```
        printf("i = %d¥n", i);
```

```
    }
```

```
    return 0;
```

```
}
```