

演習 1

次のプログラムの出力結果を、右に記入せよ

```
#include <stdio.h>

int main()
{
    int data[10];
    int i;

    data[0]=1;
    for(i=1;i<10;i++)
    {
        data[i]=data[i-1]*2;
        printf("data[%d]=%d¥n", i, data[i]);
    }

    return 0;
}
```

演習 2

次のプログラムの出力結果を、右に記入せよ

```
#include <stdio.h>

int main()
{
    int data[10]={100, 76, 83, 75, 59, 99, 40, 29, 30, 80};
    int i;

    for(i=0;i<10;i++)
    {
        if(data[i]<60)
        {
            printf("%d: 不合格¥n", i+1);
        }
        else
        {
            printf("%d: 合格¥n", i+1);
        }
    }

    return 0;
}
```

演習 3

次のプログラムの出力結果を、下に記入せよ

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

```
int main()
{
    int data[10]={100, 76, 83, 75, 59, 99, 40, 29, 30, 80};
    int i, j, temp;

    for (i=0; i<9; i++)
    {
        for (j=i+1; j<10; j++)
        {
            if (data[i]<data[j])
            {
                temp=data[j];
                data[j]=data[i];
                data[i]=temp;
                printf("%dと%dを交換\n", data[j], data[i]);
            }
        }
        for (j=0; j<10; j++)
        {
            printf("%d ", data[j]);
        }
        printf("\n\n");
    }

    return 0;
}
```

演習 4

次のプログラムは、一年生から四年生までの学齢データを与えて、それぞれの人数を計算するプログラムである。空欄を補って、プログラムを完成させよ。

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

```
int main()
{
    int gakurei [①_____]= {1, 3, 2, 4, 1, 2, 4, 2, 1, 3};
    int cnt[4]={0};
    int i;

    for (i=0; i<10; i++)
    {
        cnt[②_____ -1]++;
    }
    for (i=0; i<4; i++)
    {
        printf("%d年生は%d人です。¥n", ③_____, cnt[i]);
    }

    return 0;
}
```

演習 5

演習 4 で作成したプログラムを、不定数のデータを対象としたものへ改良せよ。具体的には、入力部分を新たに作成する。また、その部分では、1～4の値を入力するが、それ以外の値、とくに0を入力されたら入力工程を終了するなどの対応を取る。

具体的な仕様は、作成者に一任する。ある程度、不定の入力数に対応できていれば正解とする。

演習 6

aとbに対する最大公約数の求め方は以前の演習で行った。

今回はこれを拡張、aとbを配列で与え、a[i]とb[i]に対する最大公約数をc[i]に保存するプログラムを作れ。なお、プログラムの雛形は次の通りとする。必要に応じて、変数の宣言等を追加してよい。

```
#include <stdio.h>
#define MAXNUM 10

int main()
{
    int a[MAXNUM]= {126, 264, 192, 64, 253, 48, 98, 117, 223, 583};
    int b[MAXNUM]= { 72,  4, 96, 26,  7, 22, 26, 36, 44, 76};
    int c[MAXNUM]; // 18,  4, 96, 2,  1, 2, 2,  9,  1, 1
    //必要なら変数宣言を追加してよい

    for (i=0; i<MAXNUM; i++)
    {
        //この中で最大公約数を求める計算をする。

    }

    //出力
    for (i=0; i<MAXNUM; i++)
    {
        printf("a:%4d b:%4d の最大公約数はc:%5d\n", a[i], b[i], c[i]);
    }
    return 0;
}
```

参考：ユークリッドの互除法

- $a \geq b > 0$ とする。 $a < b$ なら入れ替え、 $a, b = 0$ なら終了。
- b が a を割り切れる（余りが0）なら、 b が最大公約数。
- a を b で割った余りを新たな b とし、 a は前の b （先の計算で代入する前の b ）の値を代入して、ひとつ上に戻る。
- 最小公倍数は、最初の a, b の積を最大公約数で割ったもの。

演習1 解答

```

data[1]=2
data[2]=4
data[3]=8
data[4]=16
data[5]=32
data[6]=64
data[7]=128
data[8]=256
data[9]=512

```

演習2 解答

```

1:合格
2:合格
3:合格
4:合格
5:不合格
6:合格
7:不合格
8:不合格
9:不合格
10:合格

```

演習3 解答

100 76 83 75 59 99 40 29 30 80

76と83を交換

83と99を交換

100 99 76 75 59 83 40 29 30 80

76と83を交換

100 99 83 75 59 76 40 29 30 80

75と76を交換

76と80を交換

100 99 83 80 59 75 40 29 30 76

59と75を交換

75と76を交換

100 99 83 80 76 59 40 29 30 75

59と75を交換

100 99 83 80 76 75 40 29 30 59

40と59を交換

100 99 83 80 76 75 59 29 30 40

29と30を交換

30と40を交換

100 99 83 80 76 75 59 40 29 30

29と30を交換

100 99 83 80 76 75 59 40 30 29

演習4 解答

- ① 10
- ② gakurei[i]
- ③ i+1

演習5 解答

省略

演習6 解答

省略

(単に以前のプログラムのa, bをx, yなどとし、それへa[i], b[i]を代入して計算する。
そして、その結果を出力せずに、c[i]へ代入するだけである。)