

1 流れの基本の確認

```
#include <stdio.h>
void main() {
    for( i = 0 ; i < 2 ; i++ ) {
        for( j = 2 ; j > 0 ; j-- ) {
            if ( i >= j )
                break;
            else
                printf( "%d" , i );
        }
        printf( "\n" );
    }
}
```

左のプログラムの下線 A ~ J で示された式の実行順序を示せ。この時、変数値の変化が分かるようにせよ。

	A	B	...
i	0		...
j			...
条件		真	...

2 構造体の基礎

```
#include <stdio.h>
struct Points {
    char name[ 20 ] ;
    int math ;
    int sci ;
} ;
struct Points data[] = {
    "t-saitoh" , 85 , 34 ,
    "aoyama" , 99 , 89 ,
    "sokomata" , 59 , 79 ,
} ;
void main() {
    /* この部分を作成せよ */
}
```

上記のプログラムの様な、構造体のデータがある。このデータについて、3人全員のデータについての、

1. 数学 (要素名math) については、最高点を
2. 理科 (要素名sci) については、平均点を (小数点以下も正しく求めること)

求めるプログラムを作成せよ。

3 関数との値の受渡し

XY座標を入力したら、その極座標を表示するプログラムを作成する。ただし、極座標への変換は、関数によって行う。以下のプログラムの下線部や未完成部分を完成せよ。

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

----- xy_rth( ----- , ----- ;
                double xx , double yy )
{
    ----- = ----- ; /* 絶対値を求める */
    ----- = atan2( xx , yy ) ; /* 偏角を求める */
}

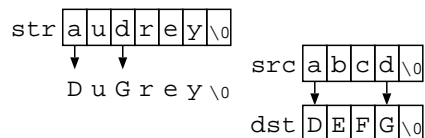
void main() {
    double x , y , r , th ;

    while( scanf( "%lf %lf" , &x , &y ) == 2 ) {
        xy_rth( ----- , ----- , x , y ) ;
        printf( "%lf %lf\n" , r , th ) ;
    }
}
```

4 文字列処理

文字列 `str[]` の中の文字を、`src[]` の中から探し、対応する `dst[]` の中の文字に置き換えたい。

プログラム中の下線部を埋めて、正しく動作するプログラムを作成せよ。



```
#include <stdio.h>

char* str_tr( char str[] , char src[] , _____ ) {
    char *s , *t , *d ;
    for( s = str ; *s != '\0' ; ___ ) {
        for( t = src , _____ ; *t != '\0' ; ___ , ___ ) {
            if ( _____ ) {
                _____ ;
                break ;
            }
        }
    }
    return str ;
}

void main() {
    /* 図の例の場合 : "DuGrey" と表示される */
    printf( "%s\n" , str_tr( "audrey" , "abcd" , "DEFG" ) ) ;

    /* rot13 と呼ばれる最も単純な暗号化 */
    /* "GSRH RH Z KVM" と表示される */
    printf( "%s\n" ,
        str_tr( "this is a pen" ,
            "abcdefghijklmnopqrstuvwxyz" ,
            "ZYXWVUTSRQPONMLKJIHGFEDCBA" ) ) ;
}
```

5 ポインタの基本の確認

```
#include <stdio.h>
struct Person {
    char   name[ 10 ] ;
    int    age ;
    double height ;
} ;
struct Person table[] = {
    { "t-saitoh" , 35 , 172.3 } ,
    { "tomoko" , 23 , 149.5 } ,
    { "mitsuki" , 0 , 68.2 } ,
    { "" , 0 , 0.0 }
} ;
double person_print( struct Person* ptr ) {
    printf( "%s , %d\n" , ptr->name , (*ptr).age ) ;
    return ptr->height ; ~~~~~A ~~~~~B
}
void main() {
    struct Person* p ;
    double s = 0.0 ;
    for( p = table ; p->name[0] != '\0' ; p++ ) {
        s += person_print( p ) ;
        ~~~~~D
    }
    printf( "%5.1f" , s ) ;
}
```

1. 左のプログラムの表示結果を示せ。
2. プログラム中の下線部 A ~ D のデータの型を具体的に述べよ。