



# 2-3 주차 실습



# 실습 1



## 배열에 저장되어 있는 최대 값 / 최소 값 구하기

- 배열 값, `int num[] = {8, 54, 11, -45, 43, 26, 66, 12, 13, 65};`
- max / min 사용자 함수를 만들어서 작성

```
#include <stdio.h>
int min(int num[], int array_size);
int max(int num[], int array_size);
void main() {
    int num[] = { 8, 54, 11, -45, 43, 26, 66, 12, 13, 65 };
    int array_size = ...;
    int i;
    printf("배열 값 :");
    printf("\n최대값 : %d\n", max(num, array_size));
    printf("최소값 : %d\n", min(num, array_size));
}
int max(int num[], int array_size) {
    ...
}
int min(int num[], int array_size){
    ...
}
```

```
배열 값 :8 54 11 -45 43 26 66 12 13 65
최대값 : 66
최소값 : -45
```

# 실습 2



## 문자열 처리 프로그램

1. 소문자를 대문자로 변환
2. 대문자를 소문자로 변환
3. 숫자로 된 문자를 숫자로 변환 후 300과 더하기
4. 입력한 문자를 거꾸로 출력

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main() {
    char input_str[100] = { "W0" };    char result_str[100] = { "W0" };
    int num;    int length, i;    int str_to_int;
    printf("사용할 문자열을 입력하시오 : ");
    gets(input_str);
    length = strlen(input_str);
    while (1) {
        printf("----- 문자열 처리 프로그램 -----Wn");
        printf("1. 소문자 to 대문자                Wn");
        printf("2. 대문자 to 소문자                Wn");
        printf("3. 문자 to 숫자 후 300과 더하기        Wn");
        printf("4. 문자를 거꾸로 출력                Wn");
        printf("0. 프로그램 종료                Wn");
        printf("-----Wn");
```

```
        printf("원하시는 처리를 숫자로 입력 해주세요:");
        scanf_s("%d", &num);
        fflush(stdin);
        switch (num) {
            ...
        }
    }
}
```

# 실습 2



## 소문자 대문자로 변환 참고 아스키 코드

문자	10진수	문자	10진수	문자	10진수	문자	10진수
A	65	N	78	a	97	n	110
B	66	O	79	b	98	o	111
C	67	P	80	c	99	p	112
D	68	Q	81	d	100	q	113
E	69	R	82	e	101	r	114
F	70	S	83	f	102	t	115
G	71	T	84	g	103	u	116
H	72	U	85	h	104	v	117
I	73	V	86	i	105	w	118
J	74	W	87	j	106	x	119
K	75	X	88	k	107	y	120
L	76	Y	89	l	108	z	121
M	77	Z	90	m	109		



## 숫자 야구 게임(Baseball)

- 0~9 까지 임의의 숫자 3개를 생성하여 맞추는 게임.
- 스트라이크(S), 볼(B), 아웃(O) 이 있음.
- 스트라이크(S) : 숫자와 위치가 모두 맞을 경우
- 볼(B) : 숫자는 맞지만 위치가 다를 경우
- 아웃(O) : 숫자, 위치 모두 맞지 않을 경우

ex) 답이 2 3 4일 경우.

1 2 3 : S(0), B(2), O(1)

4 3 2 : S(1), B(2), O(0)

5 6 7 : S(0), B(0), O(3)

2 3 4 : S(3), B(0), O(0)

### 프로그램 단계...

1. 0~9까지의 랜덤 숫자 3개 생성

(조건: 숫자가 같으면 안됨)

2. 6번의 기회를 주기 위한 반복문(for 또는 while 사용)

3. 답변자가 같은 숫자는 입력하지 못하도록 예외처리 작성

4. 10이상 입력 할 경우를 대비하는 예외처리 작성

5. for 문과 if 문을 이용하여 스트라이크, 볼, 아웃 결정하는 소스코드 작성

# 수업 실습 (다항식)



## 다항식 (다항식의 0이 아닌 항만을 배열에 저장)

```
#include <stdio.h>
#include <stdlib.h>
#define MAX_TERMS 101
typedef struct {
    float coef;
    int expon;
} polynomial;
polynomial terms[MAX_TERMS] = { { 8,3 }, { 7,1 }, { 1,0 }, { 10,3 }, { 3,2 }, { 1,0 } };
int avail = 6;
char compare(int a, int b){
    if (a > b) return '>';
    else if (a == b) return '=';
    else return '<';
}
void attach(float coef, int expon){
    if (avail > MAX_TERMS) {
        fprintf(stderr, "항의 개수가 너무 많음\n");
        exit(1);
    }
    terms[avail].coef = coef;
    terms[avail].expon = expon;
    avail++;
}
```

# 수업 실습 (다항식)



## 다항식 (다항식의 0이 아닌 항만을 배열에 저장)

```
void poly_add2(int As, int Ae, int Bs, int Be, int* Cs, int* Ce){
    float tempcoef;      *Cs = avail;
    while (As <= Ae && Bs <= Be)
        switch (compare(terms[As].expon, terms[Bs].expon)) {
            case '>': // A의 차수 > B의 차수
                attach(terms[As].coef, terms[As].expon);
                As++;break;
            case '=': // A의 차수 == B의 차수
                tempcoef = terms[As].coef + terms[Bs].coef;
                if (tempcoef)
                    attach(tempcoef, terms[As].expon);
                As++; Bs++;break;
            case '<': // A의 차수 < B의 차수
                attach(terms[Bs].coef, terms[Bs].expon);
                Bs++;break;
        }
    for (; As <= Ae; As++)
        attach(terms[As].coef, terms[As].expon);
    for (; Bs <= Be; Bs++)
        attach(terms[Bs].coef, terms[Bs].expon);
    *Ce = avail - 1;
}
```

```
void print_poly(int s, int e){
    for (int i = s; i < e; i++)
        printf("%3.1fx^%d + ", terms[i].coef, terms[i].expon);
    printf("%3.1fx^%d\n", terms[e].coef, terms[e].expon);
}
```

# 수업 실습 (다항식)



## 다항식 (다항식의 0이 아닌 항만을 배열에 저장)

```
int main(void){
    int As = 0, Ae = 2, Bs = 3, Be = 5, Cs, Ce;
    poly_add2(As, Ae, Bs, Be, &Cs, &Ce);
    print_poly(As, Ae);
    print_poly(Bs, Be);
    printf("-----\n");
    print_poly(Cs, Ce);
    return 0;
}
```

결과:

$$8x^3 + 7x + 1$$

$$10x^3 + 3x^2 + 1$$

$$18x^3 + 3x^2 + 7x + 2$$

8	7	1	10	3	1						
3	1	0	3	2	0						



# 실습 4



2x2 배열과 2x1 배열 곱을 구하여라

- A 배열, 2x2 =  $\begin{vmatrix} 5 & 7 \\ 3 & 2 \end{vmatrix}$
- B 배열, 2x1 =  $\begin{vmatrix} 2 \\ 4 \end{vmatrix}$

```
#include <stdio.h>

void main(){
    int A[2][2] = { {5,7}, {3,2}};
    int B[2][1] = { {2}, {4} };
    int result[2][1], sum, i, j, k;

    ..

}
```

# 실습 5

```
좌 상단 좌표 입력(x y): 0 0
우 하단 좌표 입력(x y): 4 4
잘못 입력하셨습니다.
우 하단 좌표 입력(x y): 4 -4
입력하신 좌표의 사각형 넓이: 16
[ 0, 0] [ 4, 0]
[ 0, -4] [ 4, -4]
```



## 구조체 실습

- x, y 좌표를 받는 구조체 생성
- 왼쪽 상단과 오른쪽 하단 좌표를 받는 사각형 구조체 생성
- 좌 상단 / 우 하단 좌표 값을 입력 받아서 사각형 좌표 값을 그리고 이에 대한 넓이 및 둘레를 구하는 프로그램 작성

Hint. 구조체 2개 생성

1. x, y 좌표를 받는 구조체,
2. 좌 상단, 우 하단을 1번 구조체를 활용하여 정의
3. 좌 상단을 int 값으로 입력 받고 while 문을 돌면서 우 하단 좌표 값을 입력 (이 때, 좌 상단 & 우 하단에 대한 조건 예외 처리 필요)
4. 너비 / 높이 값을 계산하여 넓이 구함
5. 받은 좌표 값 / 넓이 값 출력