

## <배열>

1. 사용자로부터 5개의 정수를 입력받아서 1차원 배열에 저장한다. 입력받은 배열원소를 출력하고, 또 역순으로 출력해보자.

```
#include <stdio.h>

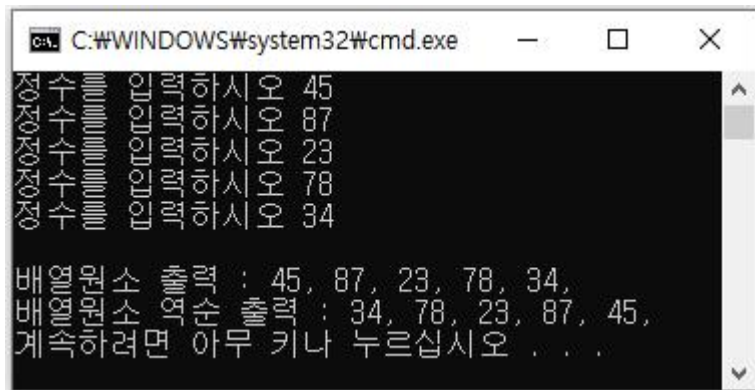
#define ST_SIZE 5

int main()
{
    int score[ST_SIZE];    //배열 선언
    int i, sum = 0;

    for ( _____ )
    {
        printf("정수를 입력하시오 ");
        scanf("%d", &score[i]);
    }
    //입력 및 누적

    printf("\n배열원소 출력 : ");
    for ( _____ )
    {
        _____;
    }
    printf("\n배열원소 역순 출력 : ");
    for ( _____ )
    {
        _____;
    }
    printf("\n");
    return 0;
}
```

실행결과 =>



```
C:\WINDOWS\system32\cmd.exe
정수를 입력하시오 45
정수를 입력하시오 87
정수를 입력하시오 23
정수를 입력하시오 78
정수를 입력하시오 34

배열원소 출력 : 45, 87, 23, 78, 34,
배열원소 역순 출력 : 34, 78, 23, 87, 45,
계속하려면 아무 키나 누르십시오 . . .
```

2. 정수형 배열 score 배열에 점수를 입력받아 저장한 후, 입력된 점수와 점수 평균을 출력하는 프로그램을 작성하려고 한다. 입력될 건수는 매크로 상수로 다음과 같이 선언되어 있다.

```
#define ST_SIZE 5
```

- 조건 1. 점수가 저장될 배열은 매크로 상수의 크기로 선언된다.
- 조건 2. 점수를 입력할 때 배열에 저장하고 누적한다.
- 조건 3. 입력이 끝나면 입력된 데이터와 평균을 출력한다.

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

```
#define ST_SIZE 5
```

```
int main()
```

```
{
```

```
    int _____;    //배열 선언
```

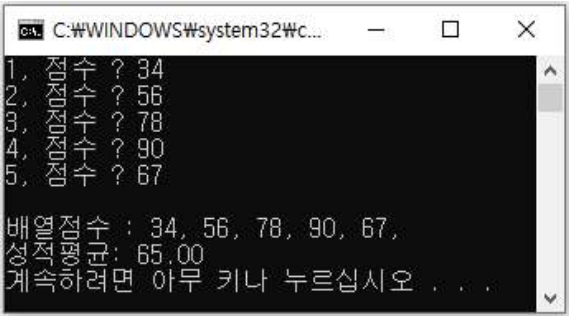
```
    int i, sum=0;
```

```
for(i=0;i<ST_SIZE;i++)
{
    _____; //입력 및 누적
    _____;
    _____;
}

printf("\n배열점수 : ");
for(i=0;i<ST_SIZE;i++)
    printf("%d, ", _____);
printf("\n성적평균: %.2f \n", _____);

return 0;
}
```

실행결과 =>

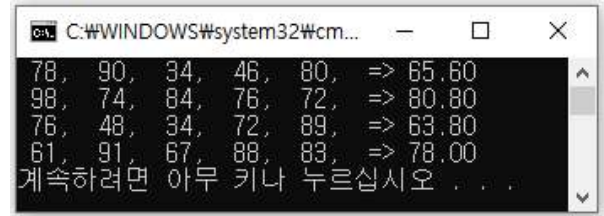


3. 다음과 같이 성적을 저장한 score 배열에 점수가 초기화되어 있다. 각 행에 있는 점수의 평균을 구하여 점수와 함께 평균을 출력하는 프로그램을 작성하시오.

- 조건 1. 성적배열 초기화  
int score[4][5] = { {78, 90, 34, 46, 80}, {98, 74, 84, 76, 72}, {76, 48, 34, 72, 89}, {61, 91, 67, 88, 83} };
- 조건 2. 행 크기, 열 크기를 계산하여 반복문에 활용한다.
- 조건 3. 각 점수 데이터와 행의 점수 평균을 출력한다.

<pre>#include &lt;stdio.h&gt;  int main() {     int score[4][5]={ {78, 90, 34, 46, 80},                       {98, 74, 84, 76, 72},                       {76, 48, 34, 72, 89},                       {61, 91, 67, 88, 83} };      int i, j, sum=0, row,col;      row=sizeof(score)/sizeof(score[0]); //행 크기     col=_____ ; //열 크기      for(i=0;i&lt;row;i++)     {         for(j=0;j&lt;col;j++)         {             _____ //데이터 출력과 누적             _____;         }         _____;         _____ //평균점수 계산 및 출력         _____;     }      return 0; }</pre>
--

실행결과 =>



4. 다음과 같이 score배열에 학생 점수가 저장되어 있다. 검색할 점수를 입력받아 score배열에서 검색하는 프로그램을 작성하시오.

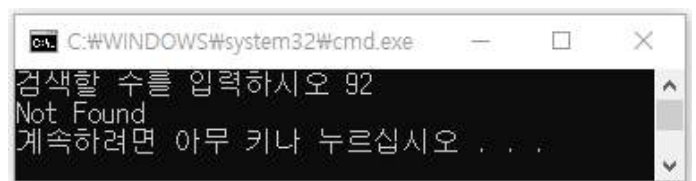
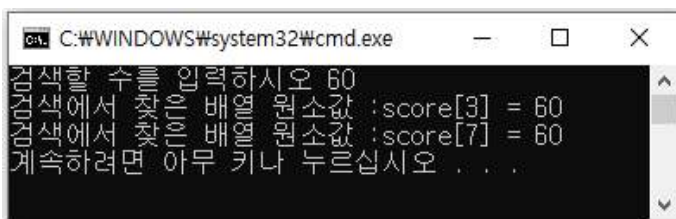
- 조건 1. 성적배열은 다음과 같이 초기화 된다.  
int score[10]={67, 93, 100, 60, 57, 86, 19, 60, 79, 84};
- 조건 2. 검색할 점수를 입력받아 score 배열에서 찾는다. 동일한 점수는 여러 개 있을 수 있다. 같은 점수는 모두 출력된다.
- 조건 3. 만약 score 배열에서 검색할 점수를 찾지 못하면 Not Found 메시지를 출력한다.

```
#include <stdio.h>

int main()
{
    int score[] = { 67, 93, 100, 60, 57, 86, 19, 60, 79, 84 };    //배열 선언
    int i, num = 0;
    _____;        //검색여부를 판단하는 변수
    _____;        //검색한 값의 위치 인덱스
    printf("검색할 수를 입력하시오 ");
    scanf("%d", &num);

    for (i = 0; i < 10; i++)
    {
        if ( _____ )
        {
            _____;        // 검색된 원소의 인덱스 값 저장
            printf("검색에서 찾은 배열 원소값 : _____");
            _____;        //검색성공을 나타냄
        }
        else
            continue;
    }
    if ( _____ )
        printf("Not Found\n");    //검색 실패
    return 0;
}
```

실행결과 =>



5. 정수형 배열 Num에 저장된 최대값이 저장된 첨자위치와 최소값이 저장된 첨자위치를 찾아 출력하는 프로그램을 작성하시오.

- 조건 1. 배열크기는 매크로 상수로 선언한다. 매크로상수는 다음과 같이 선언된다.  
#define NUM\_SZ 10
- 조건 2. 배열 Num은 다음과 같이 초기화된다.  
int Num[NUM\_SZ]={56, 900, -4, 120, 7, 90, -64, 209, 711, 10}
- 조건 3. Num배열 요소를 검색하여 최대값, 최소값이 저장된 첨자위치를 변수에 저장한다.
- 조건 4. 비교가 끝나면 최대값의 저장위치와 최소값의 저장위치와 값을 출력한다.

// 배열에 저장된 값 중에서 최대치, 최소치 구하기

```

#include <stdio.h>
#define NUM_SZ 10

int main()
{
    int Num[NUM_SZ] = { 56, 900, -4, 120, 7, 90, -64, 209, 711, 10 };
    _____;
    _____; // 최대에 해당하는 원소의 첨자를 저장할 변수
    _____; // 최소에 해당하는 원소의 첨자를 저장할 변수

    _____; // 최대치의 초기값을 첫 원소의 값으로 가정
    _____; // 최소치의 초기값을 첫 원소의 값으로 가정
    _____; // 첫 초기값의 첨자 정보를 보관
    _____; // 첫 초기값의 첨자 정보를 보관
    for (i = 1; i < NUM_SZ; i++)
    {
        if (max < Num[i])
        {
            _____;
            _____; /* max의 값이 Num[i]로 변경되었으므로
                           index도 바뀐 값의 배열 원소 첨자 값 i로 변경함*/
        }
        if (min > Num[i])
        {
            _____;
            _____; /* min의 값이 Num[i]로 변경되었으므로
                           index도 바뀐 값의 배열 원소 첨자 값 i로 변경함*/
        }
    }

    printf("==배열 원소== ");
    for (i = 0; i < NUM_SZ; i++)
        printf("%d ", Num[i]);
    printf("\n");
    printf("배열원소 최대값 : Num[%d] = %d \n", max_index, max);
    printf("배열원소 최소값 : Num[%d] = %d \n", min_index, min);

    return 0;
}

```

실행결과 =>

```

C:\WINDOWS\system32\cmd.exe
==배열 원소== 56 900 -4 120 7 90 -64 209 711 10
배열원소 최대값 : Num[1] = 900
배열원소 최소값 : Num[6] = -64
계속하려면 아무 키나 누르십시오 . . .

```

6. 3\*4행렬과 4\*3행렬을 곱해서 3\*3행렬을 구하는 프로그램을 작성하시오.

$$\begin{bmatrix} A & B & C & D \\ E & F & G & H \\ I & J & K & L \end{bmatrix} \begin{bmatrix} M & N & O \\ P & Q & R \\ S & T & U \\ V & W & X \end{bmatrix} \Rightarrow \begin{bmatrix} AM+BP+CS+DV & AN+BQ+CT+DW & AO+BR+CU+DX \\ EM+FP+GS+HV & EN+FQ+GT+HW & EO+FR+GU+HX \\ IM+JP+KS+LV & IN+JQ+KT+LW & IO+JR+KU+LX \end{bmatrix}$$

3\*4 행렬

1	2	3	4
5	6	7	8
9	10	11	12

4\*3 행렬

13	14	15
16	17	18
19	20	21
22	23	24

위 두 행렬을 곱한 3\*3행렬

1*13+2*16+3*19+4*22	1*14+2*17+3*20+4*23	1*15+2*18+3*21+4*24
5*13+6*16+7*19+8*22	5*14+6*17+7*20+8*23	5*15+6*18+7*21+8*24
9*13+10*16+11*19+12*22	9*14+10*17+11*20+12*23	9*15+10*18+11*21+12*24

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

```
int main()
```

```
{
```

```
    int mat1[3][4] = {1,2,3,4,5,6,7,8,9,10,11,12};
```

```
    int mat2[4][3] = {13,14,15,16,17,18,19,20,21,22,23,24};
```

```
    int matrix[3][3]={0};
```

```
    int i, j, sum1=0, sum2=0, sum3=0;
```

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

```
    {
```

```
        for(j=0;j<4;j++)
```

```
        {                                //행렬 계산
```

```
            _____;
```

```
            _____;
```

```
            _____;
```

```
        }
```

```
                                //배열에 합 대입
```

```
            _____;
```

```
            _____;
```

```
            _____;
```

```
        //printf("sum : %d, %d, %d \n", sum1, sum2, sum3);
```

```
        sum1=0,sum2=0,sum3=0;
```

```
    }
```

```
    //행렬 출력
```

```
    printf("3*4 mat1 행렬 \n");
```

```
    for(_____)
```

```
    {
```

```
        for(_____)
```

```

        {
            printf("%3d, ", mat1[i][j]);
        }
        printf("\n");
    }

    printf("\n4*3 mat2 행렬 \n");
    for(_____)
    {
        for(_____)
        {
            printf("%3d, ", mat2[i][j]);
        }
        printf("\n");
    }

    printf("\nmat1 과 mat2을 곱하여 3*3을 구한 matrix 행렬 \n");
    for(_____)
    {
        for(_____)
        {
            printf("%3d, ", matrix[i][j]);
        }
        printf("\n");
    }

    return 0;
}

```

실행결과 =>

```

C:\Windows\system32\cmd.exe
3*4 mat1 행렬
1, 2, 3, 4,
5, 6, 7, 8,
9, 10, 11, 12,

4*3 mat2 행렬
13, 14, 15,
16, 17, 18,
19, 20, 21,
22, 23, 24,

mat1 과 mat2을 곱하여 3*3을 구한 matrix 행렬
190, 200, 210,
470, 496, 522,
750, 792, 834,
계속하려면 아무 키나 누르십시오 . . .

```