

### 문제 1번 [숙제 목록보기]

아래와 같이 vect라는 배열에 하드코딩 해 주세요

3	5	1	1	2	3	2
---	---	---	---	---	---	---

4칸짜리 1차원 배열에 숫자 4개를 입력받아주세요

입력받은 각 숫자가 vect 배열 안에 몇개 존재하는지 출력 해 주세요

## 입력 예시

4 1 2 3

## 출력 예시

4=0개

1=2개

2=2개

3=2개

```

#include <iostream>
using namespace std;

int main(void)
{
    int vect[7] = { 3,5,1,1,2,3,2 };
    int arr[4] = {};
    for (int i = 0; i < 4; i++)
    {
        cin >> arr[i];
    }
    int arrCnt[4] = {};    // 입력받은 값이 존재하는 개수를 저장하는 배열
    for (int i = 0; i < 4; ++i)
    {
        int cnt = 0;    // 존재하는 개수
        for (int k = 0; k < 7; ++k)
        {
            if (arr[i] == vect[k])
            {
                cnt++;
            }
        }
        arrCnt[i] = cnt;
    }

    for (int i = 0; i < 4; ++i)
    {
        cout << arr[i] << "=" << arrCnt[i] << "개" << endl;
    }

    return 0;
}

```

## 문제 2번 [숙제 목록보기]

아래 배열을 하드코딩 해 주세요

10	50	40	20	30	40
----	----	----	----	----	----

이제 6칸짜리 새로운 배열을 하나 더 만들고 숫자 6개를 입력받으세요

그리고 각각의 숫자보다 큰 값이 몇개 있는지 Count해서 출력 해 주세요

만약 15 35 80 35 45 5를 입력받았다면

15	35	80	35	45	5
----	----	----	----	----	---

이렇게 배열에 값을 채우고, 하드코딩한 배열에 입력받은 값보다 큰 숫자가 몇개 존재하는지

출력 하시면 됩니다

## 입력 예시

15 35 80 35 45 5

## 출력 예시

15=5개

35=3개

80=0개

35=3개

45=1개

5=6개

```

#include <iostream>
using namespace std;

int main(void)
{
    int arr[6] = { 10, 50, 40, 20, 30, 40 };
    int arr2[6] = {};
    for (int i = 0; i < 6; ++i)
        cin >> arr2[i];

    int arrCnt[6] = {};
    for (int i = 0; i < 6; ++i)
    {
        int cnt = 0;
        for (int k = 0; k < 6; ++k)
        {
            if (arr2[i] < arr[k])
                cnt++;
        }
        arrCnt[i] = cnt;
    }

    for (int i = 0; i < 6; ++i)
    {
        cout << arr2[i] << "=" << arrCnt[i] << "개" << endl;
    }

    return 0;
}

```

### 문제 3번 [숙제 목록보기]

두 문장을 입력받아주세요 (최대 5글자)

그리고 각 문장을 정렬 합니다

정렬한 문장을 한 문장으로 옮기고 출력 해 주세요

[힌트]

가장 먼저 각 문장이 몇 글자인지 알아야 합니다

예로들어 "DCBA", "QWLBB"을 입력받았다면

문장1 정렬 : ABCD

문장2 정렬 : BBLQW

한 문장으로 만들기 : ABCDBBLQW

# 입력 예시

DCBA

QWLBB

# 출력 예시

ABCDBBLQW

```
#include <iostream>
using namespace std;

void Sort(char* str, int* num)
{
    // 문자열의 길이 구하기
    int length = 0; // 문자열의 길이
    bool isEnd = false;
    while (!isEnd)
    {
        if (str[length] == '\0')
            isEnd = true;
        else
            length++;
    }
    *num = length; // 길이 넣어주기
```

```

// 선택정렬
for (int idx = 0; idx < length; ++idx)
{
    int startX = idx + 1;
    for (int x = startX; x < length; ++x)
    {
        char target = str[idx];
        char comp = str[x];
        if (target > comp)
        {
            str[idx] = comp;
            str[x] = target;
        }
    }
}

int main(void)
{
    char str2D[2][6] = {}; // 최대 5글자
    cin >> str2D[0];
    cin >> str2D[1];
    int arrCont[2] = {};

    // 오름차순 선택정렬
    Sort(str2D[0], &arrCont[0]);
    Sort(str2D[1], &arrCont[1]);

    char strSum[11] = {};
    int idx = 0;
    for (int line = 0; line < 2; ++line)
    {
        for (int i = 0; i < arrCont[line]; ++i)
        {
            strSum[idx] = str2D[line][i];
            idx++;
        }
    }

    cout << strSum;

    return 0;
}

```

#### 문제 4번 [숙제 목록보기]

두 문장을 입력받아주세요 (최대 6글자)

두 문장이 완전히 같은 문장이라면 "같음"

다른문장이라면 "다름" 을 출력 해 주세요

[HINT] 두 문장이 모두 같은문장인지 파악하기 위해서는

일단 각 문장의 길이를 비교 해 봅니다.

조건1. 두 문장의 길이가 같은지

조건2. 다른 글자가 없는지

이 두 가지를 검사하면 됩니다.

## 입력 예시

MINS

MINST

## 출력 예시

다름

```

#include <iostream>
using namespace std;

void Counting(char* str, int* num)
{
    bool isEnd = false;
    int length = 0;
    while (!isEnd)
    {
        if (str[length] == '\0')
        {
            isEnd = true;
        }
        else
        {
            length++;
        }
    }
    *num = length;
}

int main(void)
{
    char str2D[2][7] = {};
    cin >> str2D[0] >> str2D[1];
    int arrCount[2] = {};
    bool isSame = false;

    for (int i = 0; i < 2; ++i)
    {
        Counting(str2D[i], &arrCount[i]);
    }

    if (arrCount[0] == arrCount[1])
    {
        int length = arrCount[0];
        for (int i = 0; i < length; ++i)
        {
            if (str2D[0][i] != str2D[1][i])
            {
                isSame = false;
                break;
            }
        }
    }
}

```



```

        else
        {
            isSame = true;
        }
    }
}

if (isSame)
{
    cout << "같음";
}
else
{
    cout << "다름";
}

return 0;
}

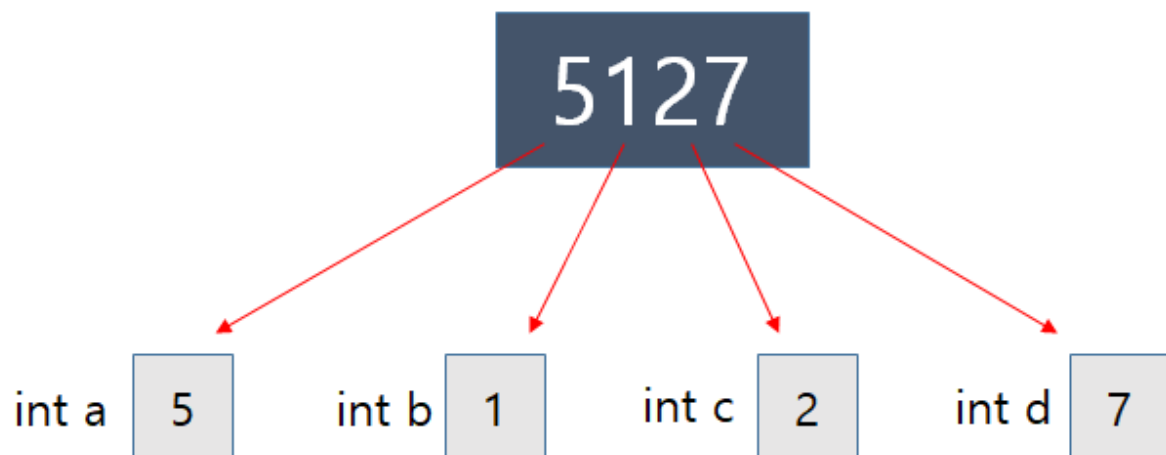
```

#### 문제 5번 [숙제 목록보기]

1000 ~ 9999 사이의 숫자 1개를 입력받아주세요

(문장으로 입력받지 않고 int형 변수에 입력 받습니다)

숫자를 쪼개서 각 자리 숫자를 4개의 변수에 옮겨담고, 각 자리를 따로 따로 출력 해주세요



[HINT] MOD 연산자를 활용해보세요

/10, %10 등등 나눗셈과 MOD연산자를 잘 활용하면

각 자릿수를 쪼갤 수 있습니다

ex) t = 13

```
int a = t / 10; //앞자리
```

```
int b = t % 10; //뒷자리
```

## 입력 예시

5127

## 출력 예시

숫자5

숫자1

숫자2

숫자7

```

#include <iostream>
using namespace std;

int main(void)
{
    int input = 0;
    while (true)
    {
        cin >> input;
        if (input >= 1000 && input <= 9999)
            break;
        else
            cout << "1000이상 9999이하의 수를 입력해주세요.\n";
    }
    int a = 0, b = 0, c = 0, d = 0;
    a = input % 10;
    input -= a;

    b = input % 100;
    input -= b;
    b /= 10;

    c = input % 1000;
    c /= 100;

    d = input / 1000;

    cout << "숫자" << a << endl;
    cout << "숫자" << b << endl;
    cout << "숫자" << c << endl;
    cout << "숫자" << d << endl;

    return 0;
}

```

### 문제 6번 [숙제 목록보기]

6명의 학생들이 일렬로 앉아있습니다.

앞사람과 뒷사람의 성적차이가 3보다 작게(<3) 차이가 나게끔 자리를 앉아야합니다

6명의 학생들 성적을 입력받고, 순차적으로 배열에 값을 넣어주세요

앞칸과 뒷칸의 성적차이가 3보다 작은지 검사하는 프로그램을 작성 해 주세요

ex) 예로들어 1 3 5 7 5 6 을 입력 받았다면

앞칸과 뒷칸이 3이상 차이나는 곳이 없으므로 "완벽한배치" 라고 출력 하세요

ex) 예로들어 2 1 6 4 3 2 를 입력 받았다면

1과 6의 차이가 3 이상 차이가 나기때문에 완벽한 배치가 아닙니다

이때는 "재배치필요" 라고 출력 해 주세요

## 입력 예시

1 3 5 7 5 6

## 출력 예시

완벽한배치

```
#include <iostream>
using namespace std;

int main(void)
{
    // 학생 6명
    // 앞, 뒤 사람의 성적차 < 3
    // 조건 만족 못하면 "재배치필요"
    // 조건 만족하면 "완벽한배치"
    int arrScore[6] = {};
    for (int i = 0; i < 6; ++i)
        cin >> arrScore[i];

    int t_idx = 0; // 비교 기준의 인덱스
    int c_idx = 0; // 비교 대상의 인덱스
    bool isPerfect = false;

    for (int i = 0; i < 5; ++i)
    {
        t_idx = i;
        c_idx = t_idx + 1;
        int sub = arrScore[t_idx] - arrScore[c_idx];
        if (sub < 0)
        {
            sub *= -1;
        }
    }
}
```

```

        if (sub > 3)
        {
            isPerfect = false;
            break;
        }
        isPerfect = true;
    }

    if (isPerfect)
        cout << "완벽한배치";
    else
        cout << "재배치필요";

    return 0;
}

```

### 문제 7번 [숙제 목록보기]

두 문장을 입력받으세요 (최대 10글자)

두 문장이 거꾸로 된 문장이 맞는지 확인하는 프로그램을 작성 해 주세요

ex) 예로들어 "ABCD" "DCBA" 를 입력받았다면

ABCD를 거꾸로 하면 DCBA가 맞습니다. 따라서 "거울문장" 이라고 출력 해 주세요

ex) 예로들어 "BKRQA" "ABC"를 입력받았다면

BKRQA를 거꾸로 하면 AQRKB 이기 때문에 "거울문장아님" 이라고 출력 해 주세요

[힌트]

변수 t를 하나 더 만들어서 활용하면 됩니다.

t = length - 1 부터 시작해서

for문 돌때마다 1씩 빼주는 방식을 쓰면 됩니다.

## 입력 예시

NCQ

QCN

## 출력 예시

거울문장

```

#include <iostream>
using namespace std;

void Counting(char* str, int* num)
{
    bool isEnd = false;
    int length = 0;
    while (!isEnd)
    {
        if (str[length] == '\0')
            isEnd = true;
        else
            length++;
    }
    *num = length;
}

int main(void)
{
    char str2D[2][11] = {}; // 최대 10글자
    cin >> str2D[0];
    cin >> str2D[1];

    int arrCount[2] = {}; // 각 문자열의 글자수 저장하는 배열
    for (int i = 0; i < 2; ++i)
    {
        Counting(str2D[i], &arrCount[i]);
    }

    bool isSame = false;
    if (arrCount[0] == arrCount[1]) // 글자수 비교
    {
        int length = arrCount[0]; // 문자열 길이
        for (int idx = 0; idx < length; ++idx)
        {
            int offset = idx + 1;
            if (str2D[0][idx] == str2D[1][length - offset])
            {
                isSame = true;
            }
            else
            {
                isSame = false;
                break;
            }
        }
    }
}

```

```

    if (isSame)
        cout << "거울문장";
    else
        cout << "거울문장아님";

    return 0;
}

```

## 문제 8번 [숙제 목록보기]

네 문장을 2차배열에 입력받아주세요 (최대 6글자)

그리고 각 문장의 길이를 저장할 4칸짜리 1차원 배열을 만들어주세요

문장의 길이를 1차원 배열에 저장한 후 정렬하고 출력하시면 됩니다

ex) 만약 "ABCD", "BBQ", "UIOUIO", "MINS" 를 입력받았다면

A	B	C	D	\0		
B	B	Q	\0			
U	I	O	U	I	O	\0
M	I	N	S	\0		

이렇게 2차배열이 채워지고 각 문장의 길이를 1차원 배열에 저장 해 주세요

4	3	6	4
---	---	---	---

이 배열을 정렬하면 3 4 4 6 이 되므로 3 4 4 6을 출력 해 주시면 됩니다

## 입력 예시

ABCD

BBQ

UIOUIO

MINS

# 출력 예시

3 4 4 6

```
#include <iostream>
using namespace std;

void Counting(char* str, int* num)
{
    bool isEnd = false;
    int length = 0;
    while (!isEnd)
    {
        if (str[length] == '\0')
            isEnd = true;
        else
            length++;
    }
    *num = length;
}

void SortNum(int* arr, int len)
{
    for (int i = 0; i < len; ++i)
    {
        int idx = i + 1;
        for (int k = idx; k < len; ++k)
        {
            if (arr[i] > arr[k])
            {
                int temp = arr[i];
                arr[i] = arr[k];
                arr[k] = temp;
            }
        }
    }
}
```



```

int main(void)
{
    char str2D[4][7] = {};    // 최대 6글자
    int arrCount[4] = {};     // 각 문장의 길이 저장
    for (int i = 0; i < 4; ++i)
    {
        cin >> str2D[i];
    }

    for (int m = 0; m < 4; ++m)
    {
        Counting(str2D[m], &arrCount[m]);
    }
    SortNum(arrCount, 4);

    for(int o = 0; o < 4; ++o)
        cout << arrCount[o] << " ";

    return 0;
}

```

#### 문제 9번 [숙제 목록보기]

한 문장을 입력받으세요 (최대 10글자)

홀수번째 문자는 대문자 이어야 합니다

짝수번째 문자는 소문자 이어야 합니다

대소문자가 번갈아가면서 나오는 문장이 맞다면 "개구리문장"

그렇지 않다면 "일반문장" 을 출력 해 주세요

## 입력 예시

AgKe0qV

## 출력 예시

개구리문장

```

int main(void)
{
    char str[11] = {};      // 최대 10글자
    cin >> str;

    bool isEnd = false;
    bool isSatisfied = false;
    int idx = 0;
    while (!isEnd)
    {
        int pos = idx + 1;
        if (str[idx] == '\0')
        {
            isEnd = true;
        }
        else if (pos % 2 == 1)    // 홀수
        {
            if (str[idx] >= 'A' && str[idx] <= 'Z')
            {
                isSatisfied = true;
            }
            else
            {
                isSatisfied = false;
                break;
            }
        }
        else if (pos % 2 == 0)    // 짝수
        {
            if (str[idx] >= 'a' && str[idx] <= 'z')
            {
                isSatisfied = true;
            }
            else
            {
                isSatisfied = false;
                break;
            }
        }
        idx++;
    }
}

```

```

    if (isSatisfied)
        cout << "개구리 문장";
    else
        cout << "일반문장";

    return 0;
}

```

### 문제 10번 [숙제 목록보기]

마을에 아래와 같은 알파벳 성을 가진 사람들이 살고 있습니다

(아래 1차원 배열을 하드코딩 해 주세요)

A	B	C	Z	E	T	Q
---	---	---	---	---	---	---

그리고 어느날, 범죄자 블랙리스트가 세상에 공개가 되었고

그 블랙리스트에 있는 사람이 마을 사람인지아닌지 확인 해 보려고합니다

블랙리스트는 다음과 같습니다

A	P	L	Q	F
---	---	---	---	---

블랙리스트 5명을 입력받아주세요

그리고 블랙리스트를 보면서 우리마을에 존재하는 사람인지 확인하는 프로그램을 작성 해 주세요

우리마을 사람이라면 "마을사람" 이라고 출력하면 됩니다

우리마을 사람이 아니라면 "외부사람" 이라고 출력하면 됩니다

## 입력 예시

APLQF

## 출력 예시

A=마을사람

P=외부사람

L=외부사람

Q=마을사람

F=외부사람

```

int main(void)
{
    char str[8] = "ABCZETQ";
    char strCrime[6] = {}; // 5명
    cin >> strCrime;
    int arrCheck[5] = {}; // 마을 사람여부 결과 저장

    for (int i = 0; i < 5; ++i)
    {
        for (int k = 0; k < 7; ++k)
        {
            if (strCrime[i] == str[k])
                arrCheck[i] = 1;
        }
    }

    for (int o = 0; o < 5; ++o)
    {
        cout << strCrime[o] << "=";
        if (arrCheck[o] == 1)
            cout << "마을사람";
        else
            cout << "외부사람";
        cout << endl;
    }

    return 0;
}

```

### 문제 11번 [숙제 목록보기]

2차배열에 다섯문장을 입력받아주세요 (최대 10글자)

이 다섯 문장 중 가장 긴 문장을 출력하시면 됩니다

[HINT]

문장의 길이를 저장 할 int length[5] 배열을 만들어주세요

이 배열에 각 문장의 길이를 저장하고 MAX 값이 있는 index를 구하면

가장 긴 문장이 어디에 있는지 찾을 수 있어요

## 입력 예시

ABC

BBQ

KFC

SHOWSHOW

OPT

## 출력 예시

SHOWSHOW

```
#include <iostream>
using namespace std;

void Counting(char* str, int* num)
{
    bool isEnd = false;
    int length = 0;
    while (!isEnd)
    {
        if (str[length] == '\0')
            isEnd = true;
        else
            length++;
    }
    *num = length;
}

int main(void)
{
    char str2D[5][11] = {}; // 최대 10글자 5문장
    for (int line = 0; line < 5; ++line)
        cin >> str2D[line];

    int arrCount[5] = {};
    int maxLine = 0;
    for (int i = 0; i < 5; ++i)
    {
        Counting(str2D[i], &arrCount[i]);
        if (arrCount[maxLine] < arrCount[i])
        {
            maxLine = i;
        }
    }
    cout << str2D[maxLine];

    return 0;
}
```