

(훈련반2) Level28.5

DFS와 BFS 문제들과 함께 나오는 내용들을 복습 해 봅니다.

Level28.5에서는 아래 내용들과 함께, For문 응용문제 훈련으로 구성되어 있습니다.

구조체 배열

Level 18의 Direct Addressing Table (DAT)

Level 19의 Direct 배열 사용

Level 20 ~ 24의 재귀호출 및 재귀호출 Trace (디버깅)

Level28.5 노드 줄세우기

문제 1번 [[숙제](#) [목록보기](#)]

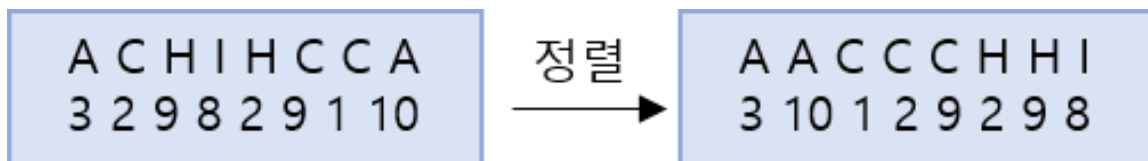
문자와 숫자로 구성된 세트를 n개 입력 받습니다.

(노드를 만들고 **구조체 배열**에 입력 받아주세요)

아래와 같은 우선순위를 기준으로, 정렬 후 결과를 출력 해주세요.

***정렬 우선순위**

1. 문자 우선
2. 같은 문자라면 숫자를 기준으로 정렬



입력 예제

```
8
A 3
C 2
H 9
I 8
H 2
C 9
C 1
A 10
```

출력 결과

A 3

A 10

C 1

C 2

C 9

H 2

H 9

I 8

```

#include <iostream>
#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtDBG.h>
using namespace std;

struct Node
{
public:
    Node()
        : num(0)
        , ch('\0')
    {
    }
    int num;
    char ch;
};

int arrSize = 0;
Node* arr = nullptr;

void init()
{
    cin >> arrSize;
    arr = new Node[arrSize]();
    for (int i = 0; i < arrSize; ++i)
        cin >> arr[i].ch >> arr[i].num;
}

void selectSort()
{
    for (int i = 0; i < arrSize - 1; ++i)
    {
        for (int k = i + 1; k < arrSize; ++k)
        {
            if (arr[i].ch > arr[k].ch)
            {
                Node temp = arr[i];
                arr[i] = arr[k];
                arr[k] = temp;
            }
            else if (arr[i].ch == arr[k].ch)
            {
                if (arr[i].num > arr[k].num)
                {
                    Node temp = arr[i];
                    arr[i] = arr[k];
                    arr[k] = temp;
                }
            }
        }
    }
}

```

```

    }
}

}

}

}

void print()
{
    for (int i = 0; i < arrSize; ++i)
        cout << arr[i].ch << " " << arr[i].num << endl;
}

void proc()
{
    selectSort();
    print();
}

void end()
{
    delete[] arr;
}

int main(void)
{
    init();
    proc();
    end();

    _CrtDumpMemoryLeaks();
    return 0;
}

```

Level28.5 재귀호출 history 출력

문제 2번 [숙제 목록보기]

Level depth와 Branch수를 입력 받으세요.

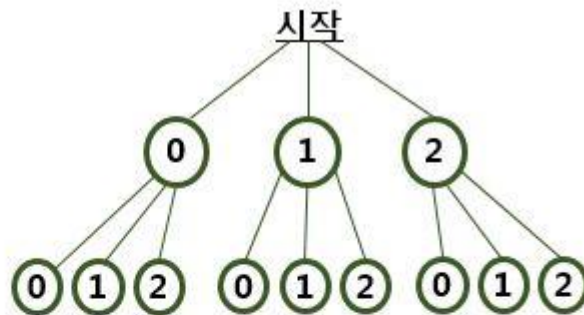
아래 그림과 같이 동작하는 재귀호출을 구현하고,

마지막 Level 에서 경로를 모두 출력 해주세요.

Ex)

Level depth = 2

자식노드 = 3



<출력결과>

```
0 0
0 1
0 2
1 0
1 1
1 2
2 0
2 1
2 2
```

입력 예제

2

3

출력 결과

```
0 0
0 1
0 2
1 0
```

1 1

1 2

2 0

2 1

2 2

```

#include <iostream>
#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtDBG.h>
using namespace std;

int levelDepth = 0;
int branchCnt = 0;
char* path = nullptr;          // 경로저장
int pathIdx = 0;
const int offset = 48;        // ascii -> '0' = 48

void init()
{
    cin >> levelDepth >> branchCnt;
    path = new char[levelDepth + 1];
    for (int i = 0; i < levelDepth + 1; ++i)
        path[i] = '\0';
}

void recursive(int _level)
{
    if (_level == levelDepth)
    {
        cout << path << endl;
        return;
    }
    for (int i = 0; i < branchCnt; ++i)
    {
        path[_level] = i + offset;
        recursive(_level + 1);
        path[_level] = '\0';
    }
}

void proc()
{
    int initLevel = 0;
    recursive(initLevel);
}

void end()
{
    delete[] path;
    path = nullptr;
}

```



```
int main(void)
{
    init();
    proc();
    end();

    _CrtDumpMemoryLeaks();
    return 0;
}
```

Level28.5 두 숫자의 합 출력

(난이도 : ★★☆☆)

문제 3번 [[숙제](#) [목록보기](#)]

6칸짜리 두 배열이 있습니다.

두배열에 6자리 숫자를 각각 채워주세요.

그리고 두 숫자의 합을 출력 해주세요.

(두 수의 합이 1,000,000보다 작은 입력값으로 주어집니다.)

자릿수 올림 처리를 잊지마세요

ex)

351429

+ 137735

489164

입력 예제

3 5 1 4 2 9

1 3 7 7 3 5

출력 결과

4 8 9 1 6 4

```

#include <iostream>
// #define _CRTDBG_MAP_ALLOC
// #include <stdlib.h>
// #include <crtdbg.h>
using namespace std;

const int arrSize = 6;
int arr1[arrSize] = {};
int arr2[arrSize] = {};
int arr3[arrSize] = {};

void init()
{
    for (int i = 0; i < arrSize; ++i)
        cin >> arr1[i];

    for (int k = 0; k < arrSize; ++k)
        cin >> arr2[k];
}

void proc()
{
    int add = 0;
    int sum = 0;
    for (int o = arrSize - 1; o >= 0; --o)
    {
        sum = arr1[o] + arr2[o];
        if (sum > 10)
        {
            arr3[o] = sum - 10 + add;
            add = 1;
        }
        else
        {
            arr3[o] = sum + add;
            add = 0;
        }
    }
}

void print()
{
    for (int i = 0; i < arrSize; ++i)
        cout << arr3[i] << " ";
}

```

```
int main(void)
{
    init();
    proc();
    print();

    //_CrtDumpMemoryLeaks();
    return 0;
}
```

Level28.5 증가하는 숫자, 감소하는 숫자 (난이도 : ★★☆☆)

문제 4번 [[숙제](#) [목록보기](#)]

5개의 숫자를 배열에 입력 받으세요.

이 숫자가 증가되는 숫자로 되어 있는지 확인하는 프로그램을 작성해주세요.

(Hint:1중 for문을 사용하면 됩니다)

증가되는 숫자로 되어 있으면 "증가됨"

아니면 "증가안됨" 이라고 출력 해주세요.

ex)

입력: 3 9 7 10 5

3	9	7	10	5
---	---	---	----	---

출력: 증가안됨

입력 예제

3 9 7 10 5

출력 결과

증가안됨

```
#include <iostream>
// #define _CRTDBG_MAP_ALLOC
// #include <stdlib.h>
// #include <crtdbg.h>
using namespace std;

const int arrSize = 5;
int arr[arrSize] = {};

int main(void)
{
    for (int i = 0; i < arrSize; ++i)
        cin >> arr[i];

    bool isAraise = true;
    for (int k = 0; k < arrSize - 1; ++k)
    {
        if (arr[k] < arr[k + 1])
        {
            isAraise = false;
            break;
        }
    }

    if (isAraise)
        cout << "증가됨";
    else
        cout << "증가안됨";

    //_CrtDumpMemoryLeaks();
    return 0;
}
```

Level28.5 숫자 분해하기 (난이도 : ★★☆☆)

문제 5번 [숙제 [목록보기](#)]

다섯 자리 숫자 1개를 입력하세요. (10000 ~ 89999 사이 숫자)

그리고 이 숫자를 분해하여 배열에 각 숫자들을 집어 넣어 주세요.

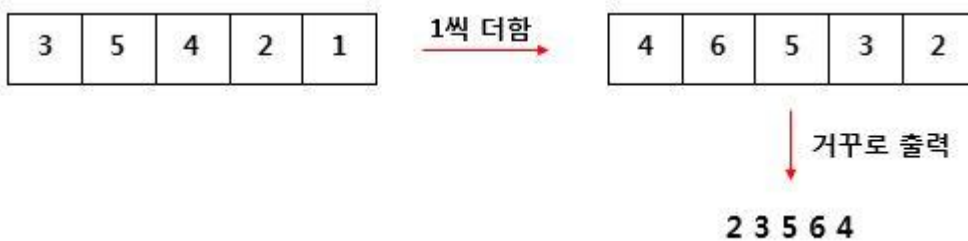
그 다음 각각 1씩 더하여 거꾸로 출력 하면 됩니다.

(자리올림을 신경쓰셔야 합니다)

ex)

입력: 35421

//띄어쓰기 없이 숫자 1개를 입력 받습니다



출력: 23564

//띄어쓰기 없이 출력하시면 됩니다

[힌트]

10 과 %10을 반복하면 숫자를 하나씩 꺼낼 수 있어요

입력 예제

35421

출력 결과

23564

```
#include <iostream>
// #define _CRTDBG_MAP_ALLOC
// #include <stdlib.h>
// #include <crtdbg.h>
using namespace std;

const int length = 5;
int input = 0;
int arr[length] = {};

int main(void)
{
    cin >> input;    // 다섯 자리 숫자

    arr[0] = input / 10000;
    input -= (arr[0] * 10000);
    arr[1] = input / 1000;
    input -= (arr[1] * 1000);
    arr[2] = input / 100;
    input -= (arr[2] * 100);
    arr[3] = input / 10;
    input -= (arr[3] * 10);
    arr[4] = input;

    for (int i = length - 1; i >= 0; --i)
        cout << arr[i] + 1;

    // _CrtDumpMemoryLeaks();
    return 0;
}
```


Level28.5 도플이니셜 찾아내기

문제 6번 [[숙제](#) [목록보기](#)]

한 문장을 입력 받으세요.(최대 10글자)

도플이니셜(같은 알파벳이 있는 문자)들을 찾고, 정렬해서 출력 해주세요.

만약 ATKPGTBA를 입력받았다면,

A와 T만 2개 이상 있으므로, 도플이니셜입니다.

ex)

ATKPGTBA → AT

입력 예제

ATKPGTBA

출력 결과

AT

```

#include <iostream>
// #define _CRTDBG_MAP_ALLOC
// #include <stdlib.h>
// #include <crtdbg.h>
using namespace std;

const int length1 = 10;
char input[length1 + 1] = {}; // 최대 10글자
// ascii -> 'A' = 65, 'Z' = 90
const int offset = 65;
const int legnth2 = 'Z' - 'A' + 1;
int isExist[legnth2] = {};

int main(void)
{
    cin >> input;
    for (int i = 0; i < length1; ++i)
    {
        int idx = input[i] - offset;
        ++isExist[idx];
    }
    for (int k = 0; k < legnth2; ++k)
    {
        if (isExist[k] >= 2)
        {
            char val = k + offset;
            cout << val;
        }
    }

    // _CrtDumpMemoryLeaks();
    return 0;
}

```

Level28.5 숫자들의 빈도수 그래프

문제 7번 [[숙제 목록보기](#)]

2차배열에 1~6까지의 숫자가 있습니다.

아래 2차배열을 하드코딩 하주세요.

그리고 숫자들의 빈도수를 그래프로 표현해주세요.(Direct Addressing Table 이용)

빈도수 그래프를 화면에 그대로 표현(출력) 해주시면 됩니다.

3	5	1
3	1	2
3	4	6
5	4	6

출력결과

```
1 * *
2 *
3 * * *
4 * *
5 * *
6 * *
```

출력 결과

```
1 * *
2 *
3 * * *
4 * *
5 * *
6 * *
```

```

#include <iostream>
// #define _CRTDBG_MAP_ALLOC
// #include <stdlib.h>
// #include <crtdbg.h>
using namespace std;

const int sizeY = 4;
const int sizeX = 3;
const int maxNum = 6;
int arr2D[sizeY][sizeX] =
{
    3,5,1,
    3,1,2,
    3,4,6,
    5,4,6,
};
int howMany[maxNum] = {};

int main(void)
{
    for (int y = 0; y < sizeY; ++y)
    {
        for (int x = 0; x < sizeX; ++x)
        {
            int idx = arr2D[y][x] - 1;
            ++howMany[idx];
        }
    }

    for (int i = 0; i < maxNum; ++i)
    {
        int starCnt = howMany[i];
        cout << i + 1 << " ";
        for (int k = 0; k < starCnt; ++k)
            cout << "* ";
        cout << endl;
    }

    // _CrtDumpMemoryLeaks();
    return 0;
}

```