



SW 역량 테스트 대비반 - 1회차

2019년 04월 11일

수업 방식

사용 언어

C/C++ (C++가 유리)

수업일시

매주 목요일
18:30 ~ 20:30

수업방식

30~40분 강의 후, 실습 수행
(7월은 실전 테스트, 필요 시 2~3개 분반 운영)

세부 커리큘럼(예상)

구분	상세 내용	교육 날짜
1주차	운영계획 및 SW 역량 레벨 테스트	04/04
2주차	C/C++ 기초 문법 강의 및 실습 1	04/11
3주차	C/C++ 기초 문법 강의 및 실습 2	04/18
4주차	기초 자료구조 강의 및 실습	04/25
5주차	정렬 알고리즘 강의 및 실습	05/09
6주차	탐색 알고리즘 강의 및 실습 1	05/16
7주차	탐색 알고리즘 강의 및 실습 2	05/23
8주차	중간 점검 모의 테스트	05/30
9주차	분할과 정복 강의 및 실습	06/07
10주차	동적 계획법 강의 및 실습	06/13
11주차	그래프 알고리즘 강의 및 실습	06/20
12주차	알고리즘 성능 향상 기법	06/27
13주차 ~	실전 감각 기르기 (레벨에 따라 2~3개 분반 필요할 수 있음)	7월 부터는 주 2회 (화/목)

개발 환경

- <https://www.onlinegdb.com/>
- 디버깅용
 - <https://visualstudio.microsoft.com/ko/vs/getting-started/mobile-install/>

익숙해지기

- 1~N까지의 합을 구해보자

INPUT	OUTPUT
10	55
100	5050
5050	12753775

익숙해지기2 (main함수 인자 전달)

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

```
int main(int argc, char *argv[])    // 옵션의 개수와 옵션 문자열을 배열로 받음
{
    for (int i = 0; i < argc; i++)    // 옵션의 개수만큼 반복
    {
        printf("arg(%d) = %s\n", i, argv[i]);
    }
    printf("argc = %d\n", argc); // argc 값 출력
    return 0;
}
```

argc == 3



익숙해지기2 (main함수 인자 전달)

- 두 수의 합을 구해보자

INPUT	OUTPUT
2 3	5
10 20	30
100 200	300


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

```
#include <stdlib.h>
```

```
int main(int argc, char *args[]){
```

```
    int num1 = atoi(args[1]);
```

```
    int num2 = atoi(args[2]);
```

```
    int sum = num1 + num2;
```

```
    printf("Sum of %d and %d is %d.",num1, num2, sum);
```

```
    return 0;
```

```
}
```

두 수의 합을 구해보자

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

int plus(int i)
{
    if (i==1) return 1;
    else return i+plus(i-1);
}

int main(int argc, char *args[]){

    int n;
    //scanf("%d", &n);
    n = atoi(args[1]);
    printf("%d\n", plus(n));
}
```

익숙해지기3 (표준 입출력 활용)

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

```
int main()
```

```
{
```

```
    char array[100];
```

```
    fgets(array, 100, stdin);
```

```
    printf("array value is %s\n", array);
```

```
    return 0;
```

```
}
```

Command line arguments:

Standard Input: ☐ Interactive Console

☒ Text

KOI

문자열 변환 (소문자 > 대문자)

(표준 입출력 활용)

INPUT	OUTPUT
ilove	ILOVE
koreA	KOREA
CODING	CODING

문자열 변환 (소문자 > 대문자)

답안

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

int main()
{
    char data[100];
    int len;
    //scanf("%s\n", data);

    fgets(data, 100, stdin);

    len = strlen(data);

    for (int i=0; i<len; i++)
    {
        data[i] = data[i] - 'a' + 'A';
    }

    printf("%s\n", data);
    return 0;
}
```

C++도 익숙해지자

0. C소스

```
#include <stdio.h>
int main(void)
{
    printf("Hello world!");

    return 0;
}
```

Hello world!



1. cout

```
#include <iostream>
using namespace std;
int main(void)
{

    cout<<"Hello world!";

    return 0;

}
```

Hello world!

C++도 익숙해지자

3. cin

```
#include <iostream>

using namespace std;

int main(void)
{
    int input;
    cin>>input;
    cout<<"input value is"<<input;

    return 0;
}
```

10

input value is 10

주의하세요!

cout의 꺾쇠 기호는 << 왼쪽을 찢습니다
cin의 꺾쇠 기호는 >> 오른쪽을 찢습니다

4. cin-문자를 입력

```
#include <iostream>

using namespace std;

int main(void)
{
    char word;

    cin>>word;

    cout<<"input word is ";
    cout<<word;

    return 0;
}
```

A

input value is A

C++도 익숙해지자

5. 서식 문자가 필요 없다

```
#include <iostream>
using namespace std;
int main(void)
{
    char name[30]={};
    char word;
    int value;
    double fvalue;

    cin>>name;
    cin>>word;
    cin>>value;
    cin>>fvalue;

    cout<<"\ninput name is ";
    cout<<name;
    cout<<"\ninput word is ";
    cout<<word;
    cout<<"\ninput number is ";
    cout<<value;
    cout<<"\ninput real number is ";
    cout<<fvalue;

    return 0;
}
```

양현식

Y

-99

3.14

input name is 양현식

input word is Y

input number is -99

input real number is 3.14

C++도 익숙해지자

7. 구구단

```
#include <iostream>

using namespace std;

int main(void)
{
    int i, j;

    for(i=1; i<10; i++)
        for(j=1; j<10; j++)
            cout << i << "★" << j << "=" << i*j<<"\n"

    return 0;
}
```

1*1=1

1*2=2

1*3=3

1*4=4

...중략...

9*8=72

9*9=81

(C++)STL을 활용한 스택 구현

스택에 1, 2, 3를 넣고..
하나씩 출력 해보자.

```
size: 3
3
2
1
empty? 1
```

```
#include <iostream>
#include <stack>
```

```
using namespace std;
```

```
int main() {
```

```
    stack<int> s;
    s.push(1);
    s.push(2);
    s.push(3);
```

```
    cout << "size: " << s.size() << endl; // 스택의 사이즈 반환
```

```
    cout << s.top() << endl; // 스택의 맨 위 반환
    s.pop(); // 스택의 맨 위 삭제
```

```
    cout << s.top() << endl; // 스택의 맨 위 반환
    s.pop(); // 스택의 맨 위 삭제
```

```
    cout << s.top() << endl; // 스택의 맨 위 반환
    s.pop(); // 스택의 맨 위 삭제
```

```
    cout << "empty? " << s.empty() << endl; // 스택이 비었는지 bool 값 반환
    return 0;
```

```
}
```

(C++)STL을 활용한 큐 구현

큐에 1, 2, 3를 넣고..
하나씩 출력 해보자.

```
#include <iostream>
#include <queue>

using namespace std;

int main() {

    queue<int> q;
    q.push(1);
    q.push(2);
    q.push(3);

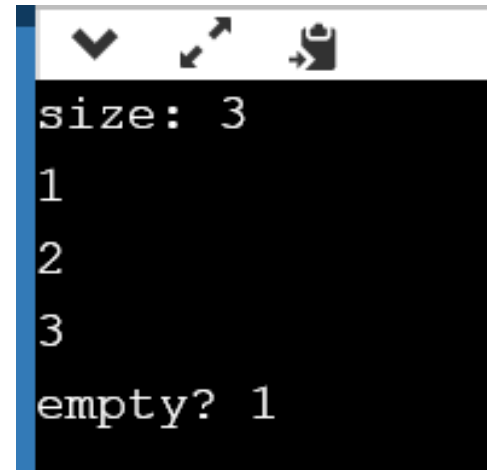
    cout << "size: " << q.size() << endl; // 큐의 사이즈 반환

    cout << q.front() << endl; // 큐의 맨 위 반환
    q.pop(); // 큐의 맨 위 삭제

    cout << q.front() << endl; // 큐의 맨 위 반환
    q.pop(); // 큐의 맨 위 삭제

    cout << q.front() << endl; // 큐의 맨 위 반환
    q.pop(); // 큐의 맨 위 삭제

    cout << "empty? " << q.empty() << endl; // 큐가 비었는지 bool 값 반환
    return 0;
}
```



```
size: 3
1
2
3
empty? 1
```

Vector 자료구조 & sort 함수

– C++ vector 자료구조 이해

<http://dream-cy.tistory.com/4>

– sort()함수 참조

<https://blockdmask.tistory.com/178>

Vector 자료구조 & sort 함수 이용 문제 풀이

- <https://programmers.co.kr/learn/courses/30/lessons/42576>

완주하지 못한 선수

문제 설명

수많은 마라톤 선수들이 마라톤에 참여하였습니다. 단 한 명의 선수를 제외하고는 모든 선수가 마라톤을 완주하였습니다.

마라톤에 참여한 선수들의 이름이 담긴 배열 participant와 완주한 선수들의 이름이 담긴 배열 completion이 주어질 때, 완주하지 못한 선수의 이름을 return 하도록 solution 함수를 작성해주세요.

제한사항

- 마라톤 경기에 참여한 선수의 수는 1명 이상 100,000명 이하입니다.
- completion의 길이는 participant의 길이보다 1 작습니다.
- 참가자의 이름은 1개 이상 20개 이하의 알파벳 소문자로 이루어져 있습니다.

입출력 예

participant	completion	return
["leo", "kiki", "eden"]	["eden", "kiki"]	"leo"
["marina", "josipa", "nikola", "vinko", "filipa"]	["josipa", "filipa", "marina", "nikola"]	"vinko"
["mislav", "stanko", "mislav", "ana"]	["stanko", "ana", "mislav"]	"mislav"

입출력 예 설명

예제 #1

"leo"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.

예제 #2

"vinko"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.

예제 #3

"mislav"는 참여자 명단에는 두 명이 있지만, 완주자 명단에는 한 명밖에 없기 때문에 한명은 완주하지 못했습니다.



```
#include <string>
#include <vector>
#include <algorithm> // sort() 사용을 위해 필요

using namespace std;

string solution(vector<string> participant, vector<string> completion) {
    string answer = "";
    sort(participant.begin(), participant.end());
    sort(completion.begin(), completion.end());
    for(int i=0;i<completion.size();i++)
    {
        if(participant[i] != completion[i])
            return participant[i];
    }
    return participant[participant.size() - 1];
    //return answer;
}
```