

C++ 프로그래밍 실습

13주차

Contents

- Exercise01~03
- Practice10
- Submit

Exercise01

```
#include <iostream>
#include <functional>
#include <algorithm>
#include <string>
#include <vector>
#include <iostream>

using namespace std;

template <typename Iter>
void print(Iter begin, Iter end){
    while(begin != end){
        cout << "[" << *begin << " ] ";
        begin++;
    }
    cout << endl;
}

struct is_odd{
    int num_delete;

    is_odd() : num_delete(0) {}

    bool operator()(const int& i){
        if(num_delete >= 2) return false;

        if(i % 2 == 1){
            num_delete++;
            return true;
        }
        return false;
    }
};

int main(){
    vector<int> vec;
    vec.push_back(5);
    vec.push_back(3);
    vec.push_back(1);
    vec.push_back(2);
    vec.push_back(3);
    vec.push_back(4);

    cout << "original vector" << endl;
    print(vec.begin(), vec.end());

    cout << "erase 2 odd elements" << endl;
    vec.erase(std::remove_if(vec.begin(), vec.end(), is_odd()), vec.end());
    print(vec.begin(), vec.end());

    return 0;
}
```

Exercise02

```
#include <iterator>
#include <vector>
#include <algorithm>
#include <iostream>
#include <random>

using namespace std;

void print(const char* msg, char a[], int len) {
    cout << msg;
    copy(a, a + len, ostream_iterator<char>(cout, " "));
}

int main(){
    const int len = 27;
    const int med = len / 2;

    vector<char> alpha;
    char alph[] = "abcdefghijklmnopqrstuvwxyz";
    //insert alphabet in vector
    for(int i=0; i<len; i++){
        alpha.push_back(alph[i]);
        //a++;
    }

    //print
    cout << "Original" << endl;
    for(int i=0; i<len; i++)
        cout << alpha[i] << " ";

    cout << endl;

    //random shuffle
    shuffle(begin(alpha), end(alpha), default_random_engine());
    cout << "Random Shuffle" << endl;
    for(int i=0; i<len; i++)
        cout << alpha[i] << " ";
    cout << endl;

    nth_element(alph, alph+med, alph+len);
    print("\n\nAfter nth_element:\n", alph, len);
    print("\n\t < median : ", alph, med);
    print("\n\t median : ", alph+med, 1);
    print("\n\t > median : ", alph+med+1, med);
    cout << endl;
    return 0;
}
```

Exercise03

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

void showWidth(){
    cout.width(10);
    cout << "Hello" << endl;

    cout.width(5);
    cout << 12 << endl;

    cout << '%';
    cout.width(10);
    cout << "Korea/" << "Seoul" << "City" << endl;
}

int main(){
    showWidth();
    cout << endl;

    cout.fill('-');
    showWidth();
    cout << endl;

    cout.precision(5);
    cout << 11. / 3. << endl;

    return 0;
}
```

Practice10

끝말잇기 게임 프로그램을 작성하시오.

```
class WordList{
private:
vector<string> dictionary;
list<string> wordlist;
public:
    WordList(); //dic.txt로부터 모든 단어들을 불러와 dictionary에 저장
    void CheckDictionary(string str); // str이 dictionary 존재 여부 확인
    void CheckDuplication(string str); // str이 wordlist 중복 여부 확인
    void CheckConfirm(string str); // str의 첫글자와 기존 리스트의 마지막 단어의 끝 글자 동일 여부 확인
    void add(string str); // 위 3가지 check 후 wordlist.push_back
    void startGame(); //게임 시작
};

int main(){
    WordList wl;
    wl.startGame();

    return 0;
}
```

- WordList() 생성자를 통해 dic.txt에 있는 모든 단어를 dictionary 자료형에 저장한다
- CheckDictionary 메소드를 통해 입력받은 단어가 사전에 있는 단어인지 확인한다.
- CheckDuplication 메소드를 통해 입력받은 단어가 기존에 나온 단어인지, 중복 여부를 확인한다.
- CheckConfirm 메소드를 통해 입력받은 단어의 앞 알파벳과 이전 단어의 끝 알파벳과 동일한지 확인한다.
- 위 3가지 메소드는 예외처리(다음 슬라이드 참조)
- Add 메소드를 통해 3가지 확인을 한 후 입력받은 단어를 wordlist에 저장하고 result.txt에도 저장한다.(다음 슬라이드 참조)
- startGame을 통해 다음과 같이 실행되도록 한다. 또한 알파벳이 아닌 입력의 경우 예외처리(다음 슬라이드 참조)

* 코드 작성 시 제시된 변수 및 메소드 외 추가적 변수 및 메소드 사용 가능. 단 제시된 변수 및 메소드 필수 사용.

Practice09

끝말잇기 게임 프로그램을 작성하시오.

```
Game Start
Input word : go
go
Input word : obtain
go obtain
Input word : net
go obtain net
Input word : nn
You Lose. Not exist word in dictionary
```

<사전에 존재하지 않는 단어인 경우>

```
Game Start
Input word : zone
zone
Input word : easy
zone easy
Input word : young
zone easy young
Input word : gene
zone easy young gene
Input word : get
You Lose. Not same as the previous word alphabet
```

<이전 단어의 마지막 글자와 일치하지 않는 단어인 경우>

```
≡ result.txt ×
Ex10 > ≡ result.txt
1 zone easy young gene |
```

<끝말잇기 단어 result.txt 파일에 저장>

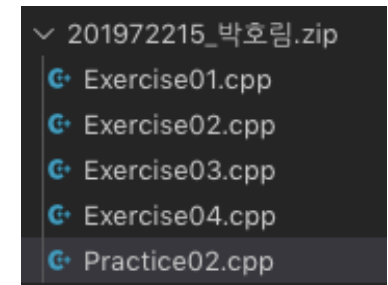
```
Game Start
Input word : up
up
Input word : proof
up proof
Input word : find
up proof find
Input word : die
up proof find die
Input word : end
up proof find die end
Input word : die
You Lose. It's Duplication
```

<중복 되는 단어인 경우>

```
Game Start
Input word : zone
zone
Input word : easy
zone easy
Input word : young
zone easy young
Input word : gene
zone easy young gene
Input word : 123
It is not English
```

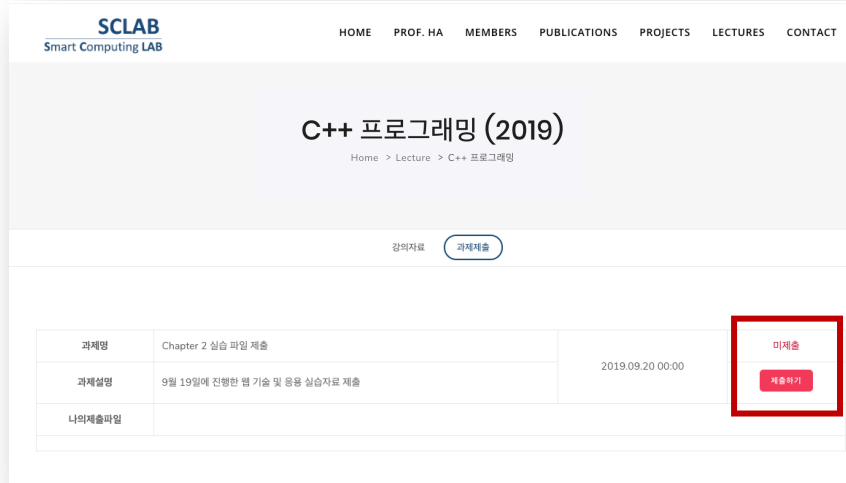
<알파벳이 아닌 입력의 경우>

- 과제
 - 실습 시간에 작성한 cpp파일들을 압축하여 보내기
 - 제출기한 : 2019.11.07(목) 자정까지
- 공통사항
 - 과제 제출 주소 : <http://sclab.konkuk.ac.kr/lecture/homework/3>
 - 과제 제출 방법
 - 홈페이지에 로그인(가입) 후 Lecture → C++ 프로그래밍 → 과제제출 페이지에서 과제 압축 후 업로드하여 제출
 - (반드시 압축하여 제출 해야하며, 제출 기한 엄수)
 - 압축 파일명 다음과 같이 지킬 것, 미 준수 시 미 채점
학번_이름.zip

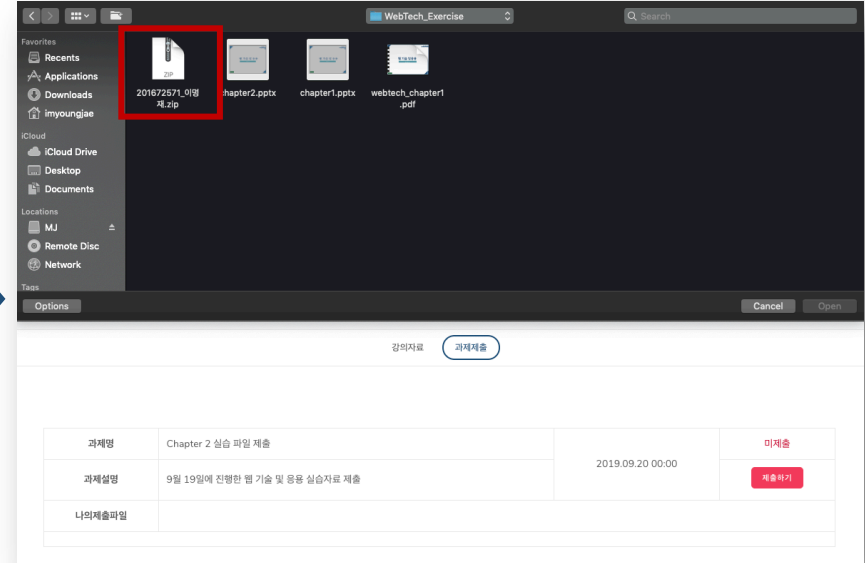


압축 예시

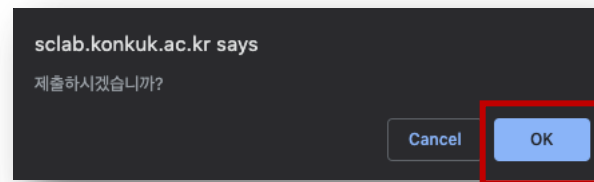
Submit



1) 제출하기 클릭



2) 제출 파일 업로드



SCLAB
Smart Computing LAB

HOME PROF. HA MEMBERS PUBLICATIONS PROJECTS LECTURES CONTACT

C++ 프로그래밍 (2019)

Home > Lecture > C++ 프로그래밍

강의자료 **과제제출**

과제명	Chapter 2 실습 파일 제출	2019.09.20 00:00	2019.09.18 15:49
과제설명	9월 19일에 진행한 웹 기술 및 응용 실습자료 제출		수정하기
나의제출파일	201672571_이명재.zip		

4) 제출 한 과제 확인 및 필요시 수정

- Teaching assistant 박호림
 - Office : 신공학관 1216호(SCLab)
 - Email : cpp2019bbaq@gmail.com
 - 수업 관련 질문이 있을 시 위의 메일로 연락 부탁드립니다.