

프로그래밍 과제 02

(6번 부터는 배열을 사용해서는 안되며, 모든 문제에서 C-string을 사용해서는 안된다.)

1. 입력으로 양의 정수들이 주어진다. 각 정수들은 8자리 이내의 정수들이다. 이 정수들을 사전식 순서 (lexicographic order)로 정렬하여 출력하는 프로그램을 작성하라. 예를 들어 입력 정수들이 75, 111, 120, 3456, 9, 12, 67, 91이라면 111, 12, 120, 3456, 67, 75, 9, 91의 순서로 출력한다. 입력은 먼저 정수의 개수 n 이 주어지고 이어서 n 개의 양의 정수들이 주어진다. 단, 숫자를 문자열(string)로 변환해서는 안 된다. 두 정수를 매개변수로 받아 첫번째 정수가 사전식 순서로 빠르면 -1, 같으면 0, 크면 1을 반환하는 함수를 구현한 후 이 함수를 이용하여 정수들을 정렬하라.

입력 예	출력
8 75 111 120 3456 9 12 67 91	111 12 120 3456 67 75 9 91
7 3467 3 78 3274 374 34213 123456	123456 3 3274 34213 3467 374 78

2. 반복적으로 한 라인의 텍스트를 입력받은 후 입력받은 텍스트에서 모든 불필요한 공백문자를 제거한 후 남은 텍스트와 그 텍스트의 길이를 다음과 같이 출력하는 프로그램을 작성하라. "exit"을 입력하면 종료한다.

입력 예	출력
This is a input text. ↵ C++ is fun ↵ exit ↵	This is a input text.:21 C++ is fun:10

3. $n \leq 100$ 개의 문자열(string)이 저장된 input3.txt파일을 읽는다. 중복된 단어들이 있을 수 있다. 입력된 단어들의 빈도수를 카운트하여 빈도수에 대한 내림차순으로 정렬하여 한 줄에 하나의 단어와 그 단어의 빈도를 출력하는 프로그램을 작성하라. 단, 빈도수가 동일한 단어들은 사전식 순서로 정렬되어야 하고, 영문 알파벳 이외의 문자는 없고, 대소문자는 구분하지 않으며 전부 소문자로 변환한다.

입력 예(INPUT3.TXT)	출력
One two three Two three four Five five FouR THREE four five FIVE four fiVe	five:5 four:4 three:3 two:2 one:1
I know what i think I Know	i:3 know:2 think:1 what:1

4. 4-125-4+77+102 등으로 표현된 수식을 문자열로 입력받아 값을 계산하는 프로그램 작성하라. 덧셈과 뺄셈만을 지원하며, 입력은 공백없는 하나의 문자열로 주어진다.

입력 예	출력
4-125-4+77+102	54
-12+3-10-9+15	-13

5. 텍스트 파일 sample_lines.txt를 읽고 파일의 라인들을 포함된 단어의 개수에 관한 내림차순으로 정렬하여 출력하는 프로그램을 작성하라. 파일의 길이는 1000라인 이내이다. 각 라인의 앞, 뒤, 단어 사이에 불필요한 공백이 있을 수 있다. 출력에서 불필요한 공백은 모두 제거되어야 한다. 공백 라인은 무시한다. 각 라인의 끝에 단어의 개수를 함께 출력하라.

입력 예	출력
What that corresponding value is depends on the importance of the variable and may also depend on where the variable is defined.	and may also depend on where:6 the importance of the variable:5 value is depends on:4 the variable is defined.:4 What that corresponding:3
five five five five five one six six six six six six three three three four four four four two two	six six six six six six:6 five five five five five:5 four four four four:4 three three three:3 two two:2 one:1

6. 텍스트 파일 sample.txt를 읽고 파일에 등장하는 단어들의 목록을 만들어 출력하는 프로그램을 작성하라. 어떤 단어가 여러 번 등장하더라도 목록에는 한 번만 나와야 하며, 단어의 앞뒤에 붙은 알파벳이나 숫자가 아닌 특수문자들(마침표, 쉼표 등)은 모두 제거하고, 모든 단어들은 소문자로 변환한다. 단어들은 사전식 순서로 정렬되어 출력되어야 한다. 파일에서 단어의 개수는 1000개를 넘지 않는다.
7. 키보드로부터 연속해서 문자열(string)들을 입력받는다. 사용자가 “exit”을 입력하면 프로그램을 종료한다. 입력에는 중복된 단어가 있을 수 있다. 하나의 단어가 입력될 때 마다 현재까지 입력된 단어들의 중복 없는 목록을 사전식 순서로 정렬하여 한 줄에 출력하는 프로그램을 작성하라. 단, 이미 목록에 있는 단어가 입력되면 단어목록 대신 “duplicate”라고 출력한다.

입력 예	출력
hello world and hello programming assignment exit	hello hello world and hello world duplicate and hello programming world and assignment hello programming world
five two five one six three six three four two exit	five five two duplicate five one two five one six two five one six three two duplicate duplicate five four one six three two duplicate

8. 먼저 양의 정수 $n < 100$ 을 입력받고 이어서 n 개의 영문 소문자 단어를 입력으로 받아 모든 문자열의 가장 긴 공통 접두사를 찾아서 출력하는 프로그램을 작성하라.

입력 예	출력
4 abcd abcdef ab abc	ab

입력 예	출력
3 xyw xyz st 6	nothing
ABCDE ABCDEF ABCDFG ABCDEF ABCDEF ABCDEF	ABC

9. shuffled_dict.txt파일에는 한 줄에 하나의 단어와 그 단어에 대한 설명이 하나의 탭 문자('t')로 구분되어 저장되어 있다. 단, 단어들은 사전식 순서로 정렬되어 있지 않다. 먼저 이 파일을 읽어서 적절한 자료구조에 단어들을 사전식 순서로 정렬하여 저장한다. 그런 다음 하나의 단어를 키보드로부터 입력받아서 사전에서 검색한다. 단어가 사전에 존재하면 그 단어와 그 단어에 대한 설명을 출력한다. 검색하는 단어가 사전에 존재하지 않으면 검색 단어가 있다고 가정했을 때 그 단어의 바로 앞에 있을 단어와 바로 다음에 있을 단어를 (그런 단어가 있다면) 단어 설명과 함께 출력한다. 또한 두 경우(존재하는 경우와 존재하지 않는 경우) 모두 검색 단어를 접두어(prefix)로 포함하는 모든 단어를 찾아서 단어와 그 단어의 설명을 출력하는 프로그램을 작성하라.

입력 예	출력
	zi Not found - zeus: the chief deity of the greeks and ruler of the upper world cf hades he was identified with jupiter + ziega: curd produced from milk by adding acetic acid after rennet has ceased to cause coagulation
zi	Results of prefix search: ziega: curd produced from milk by adding acetic acid after rennet has ceased to cause coagulation zigzag: something that has short turns or angles zillah: a district or local division as of a province zincing: the act or process of applying zinc galvanization zinnwaldite: a kind of mica containing lithium often associated with tin ore
aa	Not found + aam: a dutch and german measure of liquids varying in different cities being at amsterdam about wine gallons at antwerp at hamburg Results of prefix search: aam: a dutch and german measure of liquids varying in different cities being at amsterdam about wine gallons at antwerp at hamburg
zyn	Not found - zymome: one of a physiological group of globular bacteria which produces fermentations of diverse nature distinguished from pathogene Results of prefix search:

입력 예	출력
lee	Found: lee: that which settles at the bottom as of a cask of liquor esp wine sediment dregs used now only in the plural Results of prefix search: lee: that which settles at the bottom as of a cask of liquor esp wine sediment dregs used now only in the plural leek: a plant of the genus allium a porrum having broadly linear succulent leaves rising from a loose oblong cylindrical bulb the flavor is stronger than that of the common onion
wool	Not found - woodward: an officer of the forest whose duty it was to guard the woods + woolder: a stick used to tighten the rope in woolding Results of prefix search: woolder: a stick used to tighten the rope in woolding woolgrower: one who raises sheep for the production of wool woolsack: a sack or bag of wool specifically the seat of the lord chancellor of england in the house of lords being a large square sack of wool resembling a divan in form

10. 아래의 프로그램은 먼저 m 개의 사전식 순서로 정렬된 영문자열과, 다시 n 개의 사전식 순서로 정렬된 문자열을 입력받아 각각을 벡터 `first`와 `second`에 저장한다. 함수 `sorted_merge`는 두 벡터에 저장된 문자열을 벡터 `first`에 정렬된 상태로 합병하는 일을 한다. 이 함수를 완성하라. 아래 프로그램에서 `sorted_merge`함수의 내부를 완성하는 것 이외의 부분을 변경해서는 안된다. **라이브러리가 제공하는 `sort`함수를 사용해서는 안된다.**

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

void sorted_merge(vector<string> &first, vector<string> &second) {
    // Code here
}

int main() {
    int m, n;
    string str;
    vector<string> first, second;
    cin >> m;
    // m개의 영문자열이 사전식 순서로 정렬되어 입력된다.
    for (int i=0; i<m; i++) {
        cin >> str;
        first.push_back(str);
    }
    cin >> n;
    // n개의 영문자열이 사전식 순서로 정렬되어 입력된다.
    for (int i=0; i<n; i++) {
        cin >> str;
        second.push_back(str);
    }

    sorted_merge(first, second);

    // m+n개의 문자열이 사전식 순서로 정렬되어 출력되어야 한다.
    for (auto item: first)
        cout << item << " ";
```

```

    cout << endl;
    return 0;
}

```

입력 예	출력
5 10 13 17 19 22 4 11 15 17 23	10 11 13 15 17 17 19 22 23
8 and artist boys closest dirty essential global rotation 6 artistic broadcast close dry sense yield	and artist artistic boys broadcast close closest dirty dry essential global rotation sense yield
4 bcd cccd free glue 2 add cast	add bcd cast cccd free glue

11. 매개변수로 두 정렬된 정수 벡터 A 와 B 를 받아서 A 가 $(A \cup B) - (A \cap B)$ 가 되도록 하는 함수를 작성하라. 단, A 와 B 각각에는 중복된 정수가 없다고 가정한다. 라이브러리가 제공하는 **sort** 함수를 사용해서는 안된다. 아래 프로그램에서 `compute_set` 함수의 내부를 완성하는 것 이외의 부분을 변경해서는 안된다.

```

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

void compute_set(vector<int> &A, vector<int> &B) {
    // Code here
}

int main() {
    int m, n, k;
    vector<int> first, second;
    cin >> m;
    // m개의 정수가 오름차순으로 정렬되어 입력된다.
    for (int i=0; i<m; i++) {
        cin >> k;
        first.push_back(k);
    }
    cin >> n;
    // n개의 정수가 오름차순으로 정렬되어 입력된다.
    for (int i=0; i<n; i++) {
        cin >> k;
        second.push_back(k);
    }

    compute_set(first, second);

    for (auto item: first)
        cout << item << " ";
    cout << endl;
    return 0;
}

```

입력 예	출력
6 3 7 11 13 16 21 4 1 8 11 12 12	1 3 7 8 12 13 16 21
1 3 6 9 11 14 17 23 31 37 39 41 8 1 3 12 17 24 37 39 44	6 9 11 12 14 23 24 31 41 44
6 1 2 3 4 5 8 4 1 2 4 9	3 5 8 9

12. 사용자로부터 n 개의 정수를 입력받아 벡터에 저장한 후 크기순으로 정렬한다. 그런 다음 중복된 수를 제거하는 프로그램을 작성하라. 입력 형식은 먼저 n 의 값이 주어지고 이어서 n 개의 정수들이 주어진다. 예를 들어 $n = 8$ 이고 입력된 정수들이 4, 7, 4, 12, 4, 10, 9, 7 이라면 중복을 제거하고 남은 정수들은 4, 7, 9, 10, 12 이다. 그러면 먼저 남은 정수의 개수 5를 출력하고 콜론(:)을 출력한 후 남은 정수들을 오름차순으로 출력한다. 단, 배열을 사용해서는 안된다.

입력 예	출력
8 4 7 4 12 4 10 9 7	5: 4 7 9 10 12
5 1 1 1 1 1	1: 1

13. 입력으로 n 개의 영문 문자열을 받는다. 각 문자열의 길이는 20이하이다. 이 문자열들을 문자열의 길이가 짧은 것에서 부터 긴 것 순서로 정렬하여 출력하라. 단, 길이가 동일한 문자열들은 그들끼리 사전식 순서로 정렬되어야 한다. 입력 형식은 먼저 문자열의 개수 n 이 주어지고, 이어서 n 개의 문자열이 한 줄에 하나씩 주어진다.

입력 예	출력
6 student university school department teacher college	school college student teacher department university
12 ca ab ba cdba abcdef dcb abc c a b bcda abcd	a b c ab ba ca abc dcb abcd bcda cdba abcdef

14. 입력으로 텍스트 파일 harry.txt를 읽어서 이 파일에 등장하는 모든 단어의 목록을 중복된 단어없이 사전식 순서로 정렬하여 벡터에 저장한다. 이제 새로운 단어 하나를 키보드로부터 입력받는다. 저장된 단어들 중에 이 새로운 단어를 접두어(prefix)로 하는 모든 단어를 찾아서 한 줄에 하나씩 출력하는 프로그램을 작성하라. 마지막으로 찾아진 단어의 개수를 출력하라.

접두어의 예	출력
str	straight straightening strain strange strangely stranger strawberry stray street stretch stretched strictly striding striking striped stripes strode stroke strolled strolling strong strongest struggled 23
help	help helped helping helplessly 4
there	there therefore 2

15. 다음과 같은 형식의 입력 파일 **table.txt**를 읽어서 아래의 예와 같이 **output.txt**파일에 테이블 형태로 출력하는 프로그램을 작성하라. 입력 파일의 첫 줄에는 테이블의 행의 개수 과 열의 개수 이 주어진다. 이어진 줄에는 각 줄마다 테이블의 한 행에 들어갈 개의 내용이 문자 &로 구분되어 주어진다. 테이블의 어떤 칸은 빈 칸일 수도 있다는 것을 유념하라. 빈 칸인 경우에도 하나 이상의 공백 문자로 표현되어 있다. 불필요한 공백 문자들을 제거하여 최대한 깔끔하게 출력되도록 하라.

[입력 파일 **table.txt**의 예]

4 4 // 4행 4열의 테이블임을 의미한다.

Audio & www.ti.com/audio & Automotive and Transportation & www.ti.com/automotive
 Amplifiers & & Communications and Telecom & www.ti.com/communications
 Data Converters & dataconverter.ti.com & & www.ti.com/computers
 & www.dlp.com & Consumer Electronics & www.ti.com/consumer-apps

[출력 파일 **output.txt**의 예]

Audio www.ti.com/audio Automotive and Transportation www.ti.com/automotive
 Amplifiers Communications and Telecom www.ti.com/communications
 Data Converters dataconverter.ti.com www.ti.com/computers
 www.dlp.com Consumer Electronics www.ti.com/consumer-apps