

პროგრამირების აბსტრაქციები

სემინარის ამოცანები #4

1 ფაილებთან, სკანერთან, სტეკთან და ვექტორთან მუშაობა

ა) დაწერეთ პროგრამა, რომელიც მოცემული ტექსტური ფაილიდან წაიკითხავს სიტყვებს და მათ შებრუნებული თანმიდევრობით ჩაწერს მეორე ტექსტურ ფაილში. სიტყვების დაყოფისას შეგიძლიათ სასვენი ნიშნები არ გაითვალისწინოთ (ანუ ჩათვალოთ რომ სასვენი ნიშნები საერთოდ არ გვხვდება ტექსტში).

```
#include <iostream>
#include <fstream>
#include "console.h"
#include "tokenscanner.h"
#include "stack.h"

using namespace std;

int main() {
    TokenScanner scanner;
    Stack<string> wordStack;
    Vector<string> wordArray;

    scanner.ignoreWhitespace();

    ifstream inputFile;
    inputFile.open("input.txt");

    scanner.setInput(inputFile);

    while (scanner.hasMoreTokens())
        wordStack.push(scanner.nextToken());

    inputFile.close();

    ofstream outputFile;
    outputFile.open("output.txt");

    while (!wordStack.isEmpty()) {
        outputFile << wordStack.pop();
        if (!wordStack.isEmpty())
            outputFile << " ";
        else
            outputFile << endl;
    }

    outputFile.close();

    return 0;
}
```

ბ) წინა ნაწილში დაწერილი პროგრამა გადააკეთეთ ისე, რომ სიტყვები შებრუნებული მიმდევრობის მაგივრად სწორი მიმდევრობით იბეჭდებოდეს, მაგრამ ჯერ ყველა კენტ პოზიციაზე მყოფი სიტყვა და შემდეგ ყველა ლუწ პოზიციაზე მყოფი სიტყვა იბეჭდებოდეს.

```
#include "vector.h"

void printFromVectorStartingFromByStepInto(Vector<string>& vec,
                                           int startIndex, int step,
                                           ofstream& outputStream) {
    for (int i = startIndex; i < vec.size(); i += step) {
        outputStream << vec[i];

        if (i + step < vec.size())
            outputStream << " ";
        else
            outputStream << endl;
    }
}

int main() {
    TokenScanner scanner;
    Stack<string> wordStack;
    Vector<string> wordArray;

    scanner.ignoreWhitespace();

    ifstream inputFile;
    inputFile.open("input.txt");

    scanner.setInput(inputFile);

    while (scanner.hasMoreTokens())
        wordArray.add(scanner.next_token());

    inputFile.close();

    ofstream outputFile;
    outputFile.open("output.txt");

    printFromVectorStartingFromByStepInto(wordArray, 0, 2, outputFile);
    printFromVectorStartingFromByStepInto(wordArray, 1, 2, outputFile);

    outputFile.close();

    return 0;
}
```

2 ფაილში ლათინური ანბანის სიმბოლოებს დათვლა

მოცემული ტექსტური ფაილიდან წაიკითხეთ ტექსტი და ლათინური ანბანის თითოეული ასოსთვის დათვალეთ რამდენჯერ გვხვდება ეს ასო ტექსტში. დიდი და პატარა ასოები არ განასხვავოთ. შედეგი კონსოლში გამოიტანეთ.

```
#include <iostream>
#include <fstream>
#include "console.h"
#include "vector.h"

using namespace std;

const int alphabetSize = 26;

int main() {
    Vector<int> letterCount(alphabetSize, 0);

    ifstream inputFile;
    inputFile.open("input.txt");

    while (true) {
        char c;
        inputFile.get(c);

        if (inputFile.fail())
            break;

        if ('a' <= c && c <= 'z')
            letterCount[c - 'a']++;

        if ('A' <= c && c <= 'Z')
            letterCount[c - 'A']++;
    }

    inputFile.close();

    for (int i = 0; i < alphabetSize; i++) {
        cout << "count of " << (char)('A' + i) << " is: ";
        cout << letterCount[i] << endl;
    }

    return 0;
}
```