**1.**

#include<iostream>

#include<string>

#include<fstream>

using namespace std;

void f(string word, char first, int &n) {

if (word[0] == first)

n++;

}

int main() {

ifstream data("data.txt");

string word;

char first;

int n = 1;

data >> word;

first = word[0];

while (data >> word)

f(word, first, n);

cout << n;

}

2

#include<iostream>

#include<string>

#include<fstream>

using namespace std;

int main() {

ifstream data("data.txt");

string word, longest;

int n = 0;

while (data >> word) {

if (word[1] == 'e' && word.length() >= n) {

longest = word;

}

}

cout << longest;

}

3#include<iostream>

#include<string>

#include<fstream>

using namespace std;

void push(string \*&arr, int &n, string num) {

string\* newArr = new string[n + 1];

for (int i = 0; i < n; i++) {

newArr[i] = arr[i];

}

newArr[n] = num;

n++;

delete[] arr;

arr = newArr;

}

int main() {

ifstream data("data.txt");

string num;

int minusN = 0, plusN = 0;

string\* minus = new string[minusN];

string\* plus = new string[plusN];

while (data >> num) {

if (num[0] == '-') {

push(minus, minusN, num);

}

else {

push(plus, plusN, num);

}

}

for (int i = 0; i < plusN; i++) {

cout << plus[i] << " ";

cout << minus[i] << " ";

}

}

4

#include<iostream>

#include<string>

#include <cmath>

using namespace std;

class vect {

private:

int x = 1;

int y = 1;

public:

int getX() {

return x;

}

int getY() {

return y;

}

void setX(int value) {

x = value;

}

void setY(int value) {

y = value;

}

void showCoord() {

cout << "Coord: " << x << ", " << y << endl;

}

double modVect() {

return sqrt(x\*x + y\*y);

}

string minusVect(int x2, int y2) {

return to\_string(x-x2) + ", " + to\_string(y-y2);

}

string plusVect(int x2, int y2) {

return to\_string(x + x2) + ", " + to\_string(y + y2);

}

string multToConst(int num) {

return to\_string(x \* num) + ", " + to\_string(y \* num);

}

string multtoVect(int x2, int y2) {

return to\_string(x\*x2 + y\*y2);

}

};

int main() {

int x, y;

cin >> x >> y;

vect a;

a.setX(x);

a.setY(y);

a.showCoord();

cout << "mod " << a.modVect() << endl;

cout << "minus " << a.minusVect(2, 2) << endl;

cout << "plus " << a.plusVect(2, 2) << endl;

cout << "multToConst " << a.multToConst(2) << endl;

cout << "multToVect " << a.multtoVect(2, 2) << endl;

}