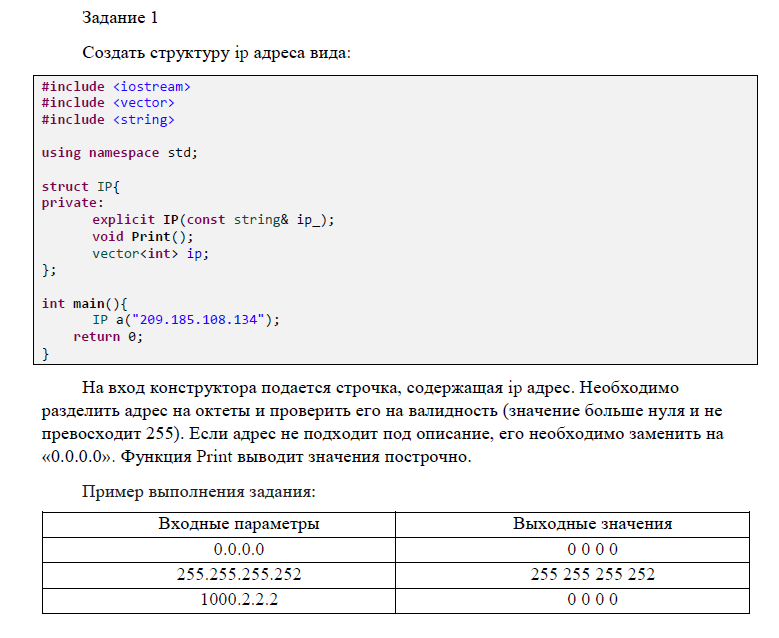
Лабораторная работа №5

Авдеев Владислав, ИКТ-31



Код:

#include <iostream>

#include <vector>

#include <string>

using namespace std;

vector<string> MegaFuncPr(string,char);

struct IP{

    public:

    explicit IP (const string& ip\_){

            vector<string> a=MegaFuncPr(ip\_,'.');

            for(auto x:a){ if (stoi(x)>=0&&stoi(x)<=255)ip.push\_back(stoi(x));

            else {ip={0,0,0,0};break;};

            }

            Print();

        };

    private:

        void Print(){for(auto x:ip) cout<<x<<' '; cout<<endl;};

        vector<int> ip;

};

int main(){

    string c;

    getline(cin,c);

    IP a(c);

    return 0;

}

vector<string> MegaFuncPr(string a,char c='.'){

    vector<string> b;

    string word="";

    for (auto x : a){

        if(x==c){

            b.push\_back(word);

            word="";

        }

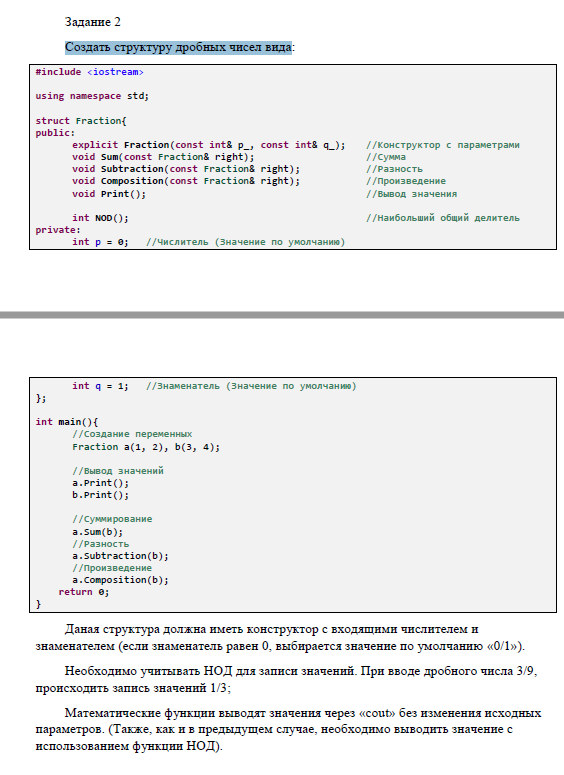
        else word+=x;

    }

b.push\_back(word);

return b;

}



Код:

#include <iostream>

using namespace std;

struct Fraction{

    public:

        explicit Fraction(const int& p\_=0, const int& q\_=1) //Конструктор с параметрами

        {

            p=p\_/NOD(p\_,q\_);

            q=q\_/NOD(p\_,q\_);

            };

        void Sum(const Fraction& right) { //Сумма

            int a,b;

            a=p\*right.q+right.p\*q;

            b=q\*right.q;

            cout<<a/NOD(a,b)<<'/'<<b/NOD(a,b)<<endl;

        };

        void Subtraction(const Fraction& right){

            int a,b;

            a=p\*right.q-right.p\*q;

            b=q\*right.q;

            if(a<0||b<0){a=abs(a); b=abs(b);cout<<'-'<<a/NOD(a,b)<<'/'<<b/NOD(a,b)<<endl;}

            else    cout<<a/NOD(a,b)<<'/'<<b/NOD(a,b)<<endl;

        };//Разность

        void Composition(const Fraction& right){

            int a,b;

            a=p\*right.p;

            b=q\*right.q;

            cout<<a/NOD(a,b)<<'/'<<b/NOD(a,b)<<endl;

        }; //Произведение

        void Print(){cout<<p<<'/'<<q<<endl;}; //Вывод значения

        int NOD(int n1,int n2){

            int div;

            if (n1 == n2)   // если числа равны, НОД найден

                return n1;

            int d = n1 - n2; // Находим разность чисел

            if (d < 0)       // если разность отрицательная,

            {

                d = -d;     // меняем знак

                div = NOD(n1, d); // вызываем функцию NOD() для двух наименьших чисел

            }

            else      // если разность n1-n2 положительная

            {

                div = NOD(n2, d); // вызываем функцию NOD() для двух наименьших чисел

            }

            return div;

        }; //Наибольший общий делитель

    private:

        int p = 0; //Числитель (Значение по умолчанию)

        int q = 1; //Знаменатель

};

int main(){

    //Создание переменных

    Fraction a(1, 2), b(3, 4);

    //Вывод значений

    a.Print();

    b.Print();

    //Суммирование

    a.Sum(b);

    //Разность

    a.Subtraction(b);

    //Произведение

    a.Composition(b);

    return 0;

}