

Решения олимпиадных задач.

2014 год. Заключительный этап. 10-11 классы. Билет 1.

Задача 1: Итерация (10)

Известно, что последовательность чисел $\{x_n\}$ ($n \geq 0$), заданная условиями

$$x_0 = \frac{a}{2}, x_{n+1} = \frac{1}{3} \left(2x_n + \frac{a}{x_n^2} \right),$$

сходится. Найти предел этой последовательности с точностью до трех знаков после запятой.

Входные данные. Во входном файле записано одно вещественное число a ($0 < a \leq 10^9$).

Выходные данные. В выходной файл вывести одно число – значение вычисленного предела с точностью до трех знаков после запятой.

Пример входного файла	Пример выходного файла
10	2.154

РЕШЕНИЕ:

```
#include "stdafx.h"
#include <math.h>

int _tmain(int argc, _TCHAR* argv[])
{
    double a, x_old, x_new;
    scanf("%lf", &a);
    x_new = a/2;
    do
    {
        x_old = x_new;
        x_new = 1.0/3.0*(2*x_old + a/x_old/x_old);
    } while (fabs(x_new - x_old) >= 0.001);
    printf("%.3lf\n", x_new);
    return 0;
}
```

Задача 2: НОК и НОД (15)

Будем обозначать через $M(a, b, c, \dots, k)$ наименьшее общее кратное, а через $D(a, b, c, \dots, k)$ — наибольший общий делитель целых чисел a, b, c, \dots, k . Вычислить

$$M(a, b, c) \cdot D(a, b) \cdot D(b, c) \cdot D(c, a).$$

Входные данные. Входной файл содержит одну строку, в которой записаны три целых числа a, b и c ($1 \leq a, b, c \leq 10^3$).

Выходные данные. В выходной файл вывести одно целое число – значение вычисленного произведения.

Пример входного файла	Пример выходного файла
9 18 45	65610

РЕШЕНИЕ:

Задачу можно решать «в лоб», но можно показать, что $\text{НОК}(a, b, c) \cdot \text{НОД}(a, b) \cdot \text{НОД}(b, c) \cdot \text{НОД}(c, a) = a \cdot b \cdot c \cdot \text{НОД}(a, b, c)$

```
#include "stdafx.h"

// Наибольший общий делитель (рекурсивная функция)
__int64 gcd(__int64 a, __int64 b)
{
    return (!b) ? a : gcd(b, a % b);
}

int _tmain(int argc, _TCHAR* argv[])
{
}
```

```

__int64 a, b, c;
scanf("%lld %lld %lld", &a, &b, &c);
printf("%lld\n", a*b*c*gcd(gcd(a, b), c));
return 0;
}

```

Задача 3: Граф (20)

Неориентированный граф G определяется как пара (V, E) , где V – непустое множество $\{v_1, v_2, \dots, v_n\}$, называемых вершинами, а E – множество неупорядоченных пар вершин, называемых рёбрами. Граф, в котором нет кратных рёбер и петель, называется простым. Путь в графе определяется как последовательность вершин, имеющая для каждой вершины ребро, соединяющее её со следующей вершиной в последовательности. Граф называется связным, если между любой парой различных вершин этого графа существует как минимум один путь. Если для графа можно указать пару различных вершин, которые не соединяются путем, то граф называется несвязным. Связный граф содержит ровно одну компоненту связности. Несвязный граф содержит более одной компоненты связности. Для заданного простого неориентированного графа, вершины которого пронумерованы числами от 1 до n , подсчитать количество компонент связности.

Входные данные. В первой строке входного файла содержится одно целое число n ($2 \leq n \leq 1000$), которое определяет число вершин в графе. Каждая из следующих n строк содержит список смежности: первое число в списке – номер текущей вершины, остальные числа – номера смежных вершин в произвольном порядке.

Выходные данные. В выходной файл вывести одно целое число – количество компонент связности.

Примеры входного файла	Примеры выходного файла
<pre> 5 1 2 3 2 1 4 3 1 4 4 2 3 5 5 4 </pre>	1
<pre> 7 1 2 2 1 3 3 2 4 4 3 5 6 7 7 6 </pre>	3

РЕШЕНИЕ:

```

#include "stdafx.h"
#include <iostream>

using namespace std;

struct edge {
    int a;
    int b[999];
    bool v;
};

edge e[1000]={0};
int n;

void f(int a)
{
    e[a].v=true;
    for (int i=0; i<e[a].a; i++)
        if (e[e[a].b[i]-1].v==false)
            f(e[a].b[i]-1);
}

int _tmain(int argc, _TCHAR* argv[])
{
    cin >> n;
    for (int i=0; i<n; i++)
    {

```

```

int c;
cin >> c;
e[c-1].v=false;
while (cin.peek() && (cin.peek()!='\r') && (cin.peek()!='\n'))
{
    cin >> e[c-1].b[e[c-1].a];
    e[c-1].a++;
}
int res=0;
for (int i=0; i<n; i++)
    if (e[i].v==false)
    {
        f(i);
        res++;
    }
cout << res << endl;
return 0;
}

```

Задача 4: Польская запись (30)

Обычный метод записи математических выражений, в которых бинарный оператор записывается между операндами, известен под названием инфиксной записи. При отсутствии скобок операции выполняются согласно правилам приоритета операторов. Для изменения порядка выполнения операций применяют скобки. Префиксная польская запись – это форма записи математических выражений, в которой оператор располагается слева от операндов. Если оператор имеет фиксированную аргументность, то в такой записи будут отсутствовать скобки, и она может быть интерпретирована без неоднозначности. Для заданного логического выражения в префиксной польской записи вычислить его значение.

Входные данные. Входной файл содержит одну строку, в которой записано логическое выражение префиксной польской записи. В исходном выражении нет пробелов, в качестве операндов используются цифры 1 (истина) и 0 (ложь), в качестве операторов используются знаки "~" (отрицание), "&" (конъюнкция), "|" (дизъюнкция). Длина каждого выражения не превосходит 100.

Выходные данные. В выходной файл вывести значение логического выражения (1 или 0).

Пример входного файла	Пример выходного файла
&&1~&1~0 10	0

РЕШЕНИЕ:

```

// Рекурсивный алгоритм вычисления префиксной польской записи

#include "stdafx.h"
#include <iostream>

using namespace std;

bool eval()
{
    char c;
    bool b1,b2;
    cin >> c;
    switch (c)
    {
        case '0': return false;
        case '1': return true;
        case '~': return !eval();
        case '&':
            b1=eval();
            b2=eval();
            return b1 && b2;
        case '|':
            b1=eval();
            b2=eval();
            return b1 || b2;
    }
}

```

```
int _tmain(int argc, _TCHAR* argv[])
{
    cout << (eval() ? '1' : '0') << endl;
    return 0;
}
```

Задача 5: Логика мышления (25)

В купе одного из вагонов Москва - Адлер ехали: москвич, минчанин, туляк, ростовчанин, харьковчанин и курянин. Их фамилии начинались буквами А, Б, В, Г, Д и Е. В дороге выяснилось, что:

- a) А и москвич – врачи, Д и минчанин – учителя, а туляк и В – инженеры.
- b) Ростовчанин, Б и Е - участники Отечественной войны, а туляк в армии совсем не служил.
- c) Харьковчанин старше А, курянин старше В, а Е – самый молодой.
- d) Б и москвич сошли в Ростове-на-Дону, а В и харьковчанин – в Армавире.

Определить начальную букву фамилии и профессию каждого из этих пассажиров.

РЕШЕНИЕ:

А	врач
Б	учитель
В	инженер
Г	инженер
Д	учитель
Е	врач