

Решения олимпиадных задач.

2014 год. Заключительный этап. 10-11 классы. Билет 2.

Задача 1: Итерация (10)

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

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

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	3.162

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 = 1.0/2.0*(x_old + a/x_old);
    } while (fabs(x_new - x_old) >= 0.001);
    printf("%.3lf\n", x_new);
    return 0;
}
```

Задача 2: НОД (15)

Будем обозначать через $D(a, b, c, \dots, k)$ – наибольший общий делитель целых чисел a, b, c, \dots, k . Вычислить

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

Входные данные. Входной файл содержит одну строку, в которой записаны четыре целых числа a, b, c и d ($1 \leq a, b, c, d \leq 10^6$).

Выходные данные. В выходной файл вывести одно целое число.

Пример входного файла	Пример выходного файла
9 18 45 4	36

РЕШЕНИЕ:

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

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

```

Задача 3: Граф (20)

Ориентированный граф G определяется как пара (X, A) , где X – непустое множество $\{x_1, x_2, \dots, x_n\}$ элементов, называемых *вершинами*, и A – семейство (a_1, a_2, \dots, a_m) элементов декартова произведения $X \times X$, называемых *дугами*. *Цепь* в графе определяется как последовательность дуг, такая, что каждая дуга в последовательности имеет один конец общий с ее предшественником, а другой конец – общий с ее последователем. *Цикл* в графе определяется как цепь, такая, что 1) никакая дуга не появляется дважды в последовательности, 2) два конца цепи суть одна и та же вершина. Для ориентированного графа без циклов, вершины которого пронумерованы числами от 1 до n , указать такой линейный порядок его вершин, что любая дуга ведет от меньшей вершины к большей. Очевидно, что если в графе есть циклы, такого порядка не существует.

Входные данные. В первой строке входного файла содержатся два целых числа n ($2 \leq n \leq 1000$) и m ($1 \leq m \leq 10000$), которые определяют число вершин и число дуг в ориентированном графе без циклов. Каждая из следующих m строк содержит два целых числа – номера двух вершин x_i и x_j ($1 \leq x_i \neq x_j \leq n$), которые соединены дугой.

Выходные данные. В выходной файл вывести последовательность номеров вершин, задающих линейный порядок. Если есть больше чем одно решение, то напечатать любое правильное.

Пример входного файла	Пример выходного файла
<pre> 9 13 1 2 1 3 1 4 2 6 2 7 3 5 4 5 4 7 5 7 6 8 7 8 7 9 8 9 </pre>	<pre> 1 4 3 5 2 7 6 8 9 </pre>

РЕШЕНИЕ:

```

// Нерекурсивный алгоритм топологической сортировки ориентированного графа без циклов
#include "stdafx.h"
#include <stdlib.h>

int find(int *b, bool *c, int n)
{
    int j, result = -1;
    for (j = 0; j < n; j++)
        if (b[j] == 0 && !c[j])
            {
                result = j;
                break;
            }
    return result;
}

int _tmain(int argc, _TCHAR* argv[])
{
    int n, m;
    scanf("%d %d", &n, &m);
    int i, j, k;
    int **a = (int **)malloc( n * sizeof(int *) );
    for (i = 0; i < n; i++)
        a[i] = (int *)malloc( n * sizeof(int) );
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)

```

```

        a[i][j] = 0;
for (k = 0; k < m; k++)
{
    scanf( "%d %d", &i, &j );
    a[i-1][j-1] = 1;
}
int *b = (int *)malloc( n * sizeof(int) );
for (j = 0; j < n; j++)
{
    b[j] = 0;
    for (i = 0; i < n; i++)
        if (a[i][j] == 1)
            b[j]++;
}
bool *c = (bool *)malloc( n * sizeof(bool) );
for (j = 0; j < n; j++)
    c[j] = false;
while ((k = find( b, c, n )) != (-1) )
{
    printf( "%4d", k+1 );
    for (j = 0; j < n; j++)
        if (a[k][j] != 0)
            b[j]--;
    c[k] = true;
}
printf("\n");
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. Постфиксная польская запись – это форма записи математических выражений, в которой операнды расположены перед оператором. Если оператор имеет фиксированную арность, то в такой записи будут отсутствовать скобки, и она может быть интерпретирована без неоднозначности. Для заданного логического выражения в постфиксной польской записи вычислить его значение.

Входные данные. Входной файл содержит одну строку, в которой записано логическое выражение в постфиксной польской записи. В исходном выражении нет пробелов, в качестве операндов используются латинские буквы T (истина) или F (ложь) в верхнем регистре, в качестве операторов используются знаки "~" (отрицание), "&" (конъюнкция), "|" (дизъюнкция). Длина каждого выражения не превосходит 100.

Выходные данные. В выходной файл вывести значение логического выражения (T или F).

Пример входного файла	Пример выходного файла
TT~F~~ &TF &	F

РЕШЕНИЕ:

```

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

bool stack[100];
int top=-1;

void push(bool a)
{
    stack[++top]=a;
}

bool pop()
{
    top--;
    return (stack[top+1]);
}

bool eval(char *s)

```

```

{
    bool a, b;
    unsigned int i;
    for (i=0; i<strlen(s); i++)
    {
        switch (s[i])
        {
            case 'F':
                push(false);
                break;
            case 'T':
                push(true);
                break;
            case '~':
                a=pop();
                push(!a);
                break;
            case '&':
                a=pop();
                b=pop();
                push(a && b);
                break;
            case '|':
                a=pop();
                b=pop();
                push(a || b);
                break;
        }
    }
    return (stack[0]);
}

int _tmain(int argc, _TCHAR* argv[])
{
    char s[101];
    scanf("%s", s);
    printf("%c\n", eval(s) ? 'T' : 'F');
    return 0;
}

```

Задача 5: Логика мышления (25)

Десять мальчиков: Александр, Борис, Василий, Георгий, Дмитрий, Евгений, Захар, Иван, Кирилл и Леонид учатся все в разных классах одной средней общеобразовательной школы. Известно, что:

- Старший брат Дмитрия оканчивает VII классов, а младший брат Евгения учится в V классе.
- Александр старше Кирилла на один класс, а Леонид старше Евгения на два класса.
- Василий не оканчивает X классов в этом году, Иван при окончании III класса получил похвальную грамоту, Борис - куратор в V классе, а Василий - в IV.
- Александр, Кирилл и шестиклассник начали сдавать нормы 2-го разряда по плаванию, а Борис, Евгений и восьмиклассник уже сдали нормы 1-го разряда по плаванию.
- Александр и семиклассник живут на Покровке, Георгий и пятиклассник – на Ильинке, первоклассник и восьмиклассник – на Петровке, а Кирилл и десятиклассник – на Сретенке.
- Борис помогает отстающему Евгению, Иван получает помощь от Дмитрия, Александр – от Георгия.

В каком классе учится каждый из них?

РЕШЕНИЕ:

Александр	3
Борис	10
Василий	8
Георгий	6
Дмитрий	5
Евгений	7
Захар	1
Иван	4
Кирилл	2
Леонид	9