

Всероссийская олимпиада школьников по информатике
Муниципальный этап

Методика проверки решений задач и описание системы оценивания
7– 8-й класс

Максимальное количество баллов, которое может набрать участник, – 500. Каждая задача оценивается, исходя из максимума в 100 баллов.

Проверка осуществляется в автоматическом режиме с использованием системы проведения соревнований «Яндекс.Контест» (contest.yandex.ru).

В случае невозможности использования системы «Яндекс.Контест» необходимо предоставить справку от провайдера об отключении Интернета, составленный акт о причинах неприменения системы и информацию об использованной автоматизированной системе и применённых методах проверки.

Задача 1. Блинная битва

100 баллов

За каждое правильно записанное в ответе число, располагающееся на своём месте, начисляется 20 баллов. В случае полного совпадения с ответом выставляется 100 баллов.

Ответы и разбор

Если игрок не может сделать ход, то позиция проигрышная. Если из позиции можно перейти в проигрышную, то она выигрышная. Если из позиции нельзя перейти в проигрышную, то она сама проигрышная. По условию задачи нужно найти проигрышную позицию, с наименьшим номером, но больше или равным N. Ниже представлено начало таблицы состояний для каждого набора ходов.

№	Ходы (К,Р)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	(1, 2)	P	W	W	P	W	W	P	W	W	P	W	W	P	W	W	P
2	(1, 3)	P	W	P	W	P	W	P	W	P	W	P	W	P	W	P	W
3	(2, 5)	P	P	W	W	P	W	W	P	P	W	W	P	W	W	P	P
4	(1, 4)	P	W	P	W	W	P	W	P	W	W	P	W	P	W	W	P
5	(3, 7)	P	P	P	W	W	W	P	W	W	W	P	P	P	W	W	W

№ задания	Условие	Ответ
1	$K = 1, P = 2, N = 5$	6
2	$K = 1, P = 3, N = 12$	12
3	$K = 2, P = 5, N = 20$	21
4	$K = 1, P = 4, N = 13$	15
5	$K = 3, P = 7, N = 23$	26

Задача 2. Обереги из рун

100 баллов

За каждое правильно записанное в ответе число, располагающееся на своём месте, начисляется 20 баллов. В случае полного совпадения с ответом выставляется 100 баллов.

Ответы и разбор

1) 625

Всего вариантов: 5^4 .

Вычисление: $5^4 = 5 \times 5 \times 5 \times 5 = 625$.

2) 369

Легче посчитать дополнение — обереги **без** буквы А, и вычесть из общего числа.

Число оберегов без буквы А: используют только 4 оставшихся руны $\Rightarrow 4^4$.

$4^4 = 4 \times 4 \times 4 \times 4 = 256$.

Всего оберегов: $5^4 = 625$.

Значит, оберегов с хотя бы одной А: $625 - 256 = 369$

3) 256

Это уже посчитано: без А — $4^4 = 256$.

4) 96

Выбрать позиции для двух букв А: 6.

В оставшихся двух позициях поставить любую из четырёх букв (не А): $4^2 = 16$.

Всего: $6 \times 16 = 96$.

5) 120

Нужно выбрать по порядку 4 разных руны из 5, сначала у нас 5 вариантов, потом остается 4, затем 3, затем 2: $5 \cdot 4 \cdot 3 \cdot 2 = 120$.

Задача 3. В поле с конём

100 баллов

За каждый правильный ответ начисляется 20 баллов. В случае полного совпадения с ответами выставляется 100 баллов.

Ответы

№ задания	Условие	Ответ
1	$X = E4, N = 1$	8
2	$X = A1, N = 2$	10
3	$X = B5, N = 4$	32
4	$X = C3, N = 11$	32
5	$X = D5, N = 150$	32

Разбор

Стоит заметить, что после каждого хода цвет клетки меняется, поэтому конь может быть только на клетках одного цвета после одного и того же количества ходов, следовательно, количество возможных клеток не более 32.

1) $X = E4, N = 1$

0 ход: 1 клетка: E4

1 ход: 8 клеток: C3, C5, D2, D6, F2, F6, G3, G5

2) $X = A1, N = 2$

0 ход: 1 клетка: A1

1 ход: 2 клетки: B3, C2

2 ход : 10 клеток:

A1, A3, A5, B4, C1, C5, D2, D4, E1, E3

3) $X = B5, N = 4$

После 0 ходов: 1 клетка: B5

После 1 хода: 6 клеток: A3, A7, C3, C7, D4, D6

После 2 ходов: 21 клетка: A2, A4, A6, A8, B1, B3, B5, B7, C2, C4, C6, C8, D1, D5, E2, E4, E6, E8, F3, F5, F7

После 3 ходов: 32 клетки, все клетки одного цвета; конь уже охватывает 32 клетки цвета, на котором он может быть после 3 ходов

После 4 ходов: снова 32 клетки одного цвета

4) $X = C3, N = 11$; 5) $X = D5, N = 150$

Когда ходов очень много, количество клеток упирается в верхнюю границу из 32 клеток одного цвета.

Задача 4. Степенная игра

100 баллов

За каждый правильный ответ начисляется 20 баллов. В случае полного совпадения с ответами выставляется 100 баллов.

Ответы

№ задания	Условие	Ответ
1	$N = 4$	3
2	$N = 14$	6
3	$N = 1024$	11
4	$N = 47293$	66
5	$N = 1000000000$	268

Разбор

Для маленьких N можно было промоделировать игру вручную. Программа для полного решения поставленной задачи может принимать, например, следующий вид:

```
#include <iostream>
using namespace std;
using ll = long long;

using namespace std;
int main(int argc, char* argv[]) {
    ll n, a, b;
    cin >> n; //считываем количество камней
    a = 0; //количество камней на предыдущем ходу
    b = 0; //количество ходов
    while (n) { //пока камни не закончатся, моделируем игру
        if (a == 0) { //если это первый ход и никто не брал камней
            a = 1;
        }
        else {
            if (a * 2 <= n) { //если можно взять удвоенное количество
                a = a * 2;
            }
            else { //иначе можно взять только один камень
                a = 1;
            }
        }
        n -= a; //берем камни из кучи
        b++; //увеличиваем счетчик ходов на один
    }
    cout << b << endl;
}
```

Задача 5. Справедливый дележ

100 баллов

Решение

Нам нужно найти количество натуральных делителей числа N , включая 1 и само число N . Почему?

Если мы раздаём N пряников K людям поровну, то каждый получит N/K пряников. Чтобы это было целое число, K должно быть делителем N . $K \geq 1$, поэтому учитываем все натуральные делители.

Простой случай ($N \leq 10^6$)

Для небольших N можно использовать прямой перебор:

- Перебираем все числа от 1 до N
- Проверяем, делится ли N на это число
- Считаем количество делителей

Сложность: $O(N)$ — приемлемо для $N \leq 10^6$

Программа, набирающая 50 баллов, может иметь подобный вид на ЯП Python:

```
n = int(input()) # считывание N
count = 0 # счётчик ответа
for i in range(1, n + 1): # перебираем от значения 1 до n
    if n % i == 0: # и если нашли делитель
        count += 1 # увеличиваем ответ
print(count) # в конце выводим
```

Сложный случай ($N \leq 10^{12}$)

Для больших N прямой перебор до N невозможен (10^{12} операций — это слишком много). Эффективный алгоритм:

- Перебираем числа только до \sqrt{N}
- Если i делит N , то мы находим сразу два делителя: i и N/i
- Особый случай: когда N — полный квадрат ($i = N/i$)

Сложность: $O(\sqrt{N})$ — приемлемо даже для $N = 10^{12}$ (всего 10^6 операций)

Из гипотетических трудностей - на больших значениях корень может вычисляться не очень-то и точно. Впрочем, ничего не мешает оставаться в целых числах.

Программа решения может принимать, например, следующий вид на ЯП Python:

```
# Чтение исходных данных
n = int(input())
# Инициализация переменных
count = 0
i = 1
# Проход до корня из n с подсчетом делителей
while i * i <= n:
    if n % i == 0:
        count += 1 if i * i == n else 2
    i += 1
# Вывод результата
print(count)
```

Программа решения может принимать, например, следующий вид на ЯП C++:

```
#include <iostream>
using namespace std;

int main() {
// Чтение исходных данных
    long long n;
    cin >> n;
// Инициализация переменных
    long long count = 0;
// Проход до корня из n с подсчетом делителей
    for (long long i = 1; i * i <= n; i++) {
        if (n % i == 0) {
            if (i * i == n) {// Если n - полный квадрат
                count += 1;
            } else {
                count += 2;
            }
        }
    }
// Вывод результата
    cout << count << endl;
    return 0;
}
```