Олимпиада школьников «Надежда энергетики»

МЭИ, г. Москва

Место проведения

BS 88-89

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ АРШИНОВ

ИМЯ ВЛАДИСЛАВ

ОТЧЕСТВО ПАВЛОВИЧ

Дата рождения 20.10.2002 **Класс:** 10

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 88 -89

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1

Только два датчика: на ~~про~~ крайнюю правую и на нижнюю правую. Рассмотрим на примерах:

Обозначим правую крайнюю за I, а правую нижнюю — II.

I + II −   I + II +   I − II +   I − II −

Примечание: I+ значит, что черта I прочерчена, I − — нет.

Мы видим, что комбинации разные, значит мы можем идентифицировать знак такими датчиками.

Меньше взять не получится, т.к. у черты два состояния: прочерчена или нет. Невозможно двумя состояниями анализировать и представлять четыре разных комбинации. +

№2.

Лучше всего будет пойти в двоичную систему счисления и представлять ~~массы~~ как гири как цифры в числе массы. Тогда нам придется взять 7 гирь массами 1, 2, 4, 8, 16, 32 и 64.

±

~~№3 нет~~?

```
table = <ввод таблицы в виде двумерного списка>
line = [] # в этот список мы запишем числа в том порядке, в каком
             они должны идти по условию
# и да, индексация в списках идёт от нуля
for i от 0 до N-1 {      # N- размер таблицы
  line += table[i][::-i%2] # здесь мы берем строку таблицы и
                             с помощью срезов разворачиваем
                             строки с нечётными индексами
}
for i от 0 до N^2-1 { # теперь сам поиск
  if d ≤ line[i] ≤ S : print(i//N, i%N) # выводим индекс строки и
                                          столбца именно в таком
                                          порядке.
```

№3?

∓

№4.

1. ~~Общая формула i-го слагаемого:~~

Формула N-го слагаемого:

$$(-1)^{N-1}\cdot\frac{x^{2N-1}}{N}$$

2. Формула из условия рекурсивная, поэтому если нам надо будет просто определить N-е слагаемое (и больше ничего), она очевидно проиграет формуле из п. 1.

Выразим из этой формулы N-е слагаемое:

$$S_N = -x^2 S_{N-1}\cdot\frac{N-1}{N}$$

Здесь пять арифметических операций:

$$-x\overset{1}{\cdot}x\overset{2}{\cdot}S_{N-1}\overset{3}{\cdot}(N\overset{4}{-}1)\overset{5}{:}N$$

Т.е. для расчета N-го слагаемого потребуется $5(N-1)$ операций. Теперь посмотрим на формулу из п. 1. a^n — всегда $(n-1)$ операций. Тогда кол-во операций в формуле п. 1:

$(N-2)+1+(2N-2)+1=3N-2$

Из это, сравним: (верное, доказали)

$3N-2 \leq 5(N-1)$

$2N \geq 3$

$N \geq 1{,}5$

То есть начиная со второго слагаемого мы будем совершать меньше арифм. операций, используя формулу из п. 1.

Но это все в той ситуации, если, повторюсь, нам надо определить N-е слагаемое и ничего более. Рассмотрим ситуацию, когда мы уже считаем саму сумму, поочередно складывая слагаемые (тавтология).

Мы считаем сумму N слагаемых. В случае с формулой из условия ничего не меняется: все те же $5(N-1)$ операций на одно слагаемое. Но в случае с формулой из п. 1 нам придется на каждой i-й итерации выполнять $3i-2$ операций. Тогда у нас получится арифметическая прогрессия с базой $3\cdot 2-2=4$, и ее сумма будет равна

$$S(N)=\frac{(4+3N-2)(N-1)}{2}=\frac{(3N+2)(N+1)}{2}$$

Здесь N уже встает в квадрат, а значит эта сумма будет однозначно больше, чем $5(N-1)$.

Напомню, только что я рассмотрел ситуацию, когда мы считаем сумму N слагаемых.

(+)

№5.

По сути, все остатки от деления на число m лежат в промежутке $[0; m)$. ~~Но есть алгоритм сводящийся к проверке на наличие у квадратного уравнения корней вообще, а затем к проверке попадания в $Z_m = [0; m)$.~~

~~a, b, c, m = <ввод>~~

~~if $b^2 \geq 4ac$ and $\left(0 \leq \frac{-b+\sqrt{b^2-4ac}}{2a} < m\right.$ or $\left.0 \leq \frac{-b-\sqrt{b^2-4ac}}{2a}\right)$: print("Радуйся, решение есть!")~~

И поскольку в условии сказано про натуральные числа, множество весьма ограничено. Самое простое решение — просто перебрать все возможные варианты:

```
a, b, c, m = <ввод>
for x от 0 до m-1 {
    if ((a×x²)%5 + bx%5 + c)%5 == 0: {print("Радуйся, Тимон, решение найдено!")
                                       break }
}
else print("Увы, решений не найдено")
```

x%y — остаток от деления x на y.

±

Олимпиада школьников «Надежда энергетики»

СФМЭИ

Место проведения

GE 28-39

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 43991

ФАМИЛИЯ Балюконис

ИМЯ Тимофей

ОТЧЕСТВО Олегович

Дата рождения 27.02.2005 Класс: 9

Предмет информатика Этап: Заключительный

Работа выполнена на 4 листах Дата выполнения работы: 16 февраля 2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

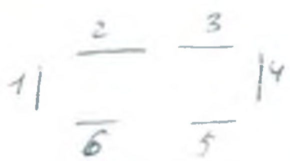
Вариант: 73991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ GE 28-39

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1

рассмотрим полный прямоугольник и все варианты:



1) сегменты 1, 2, 3 встречаются в каждом знаке, следовательно датчиков для них не требуется.

рассмотрим случаи, которые остались:

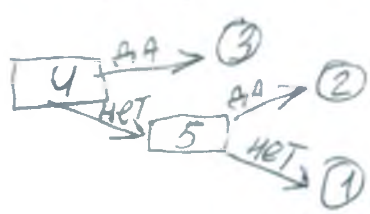
Несложно заметить, что 6 сегмент не влияет на результат.

Следовательно, нам требуется 2 датчика.

Алгоритм работы прогр.: (0 - ~~есть~~ нет полосы, 1 - есть)

если датчик 4 показывает 0, значит метка ③ (знак)

иначе если датчик 5 показывает 1, значит метка ②, иначе ①



±

№2

Нам достаточно гирь весами 1, 3, 9, 27 или всех степеней 3 до 40.

+

Пойдем от Q, ~~...~~ Составим все тройки чисел (поскольку из каждого числа можно сделать число на 1 меньше или больше). Разница между ними будет 1. Повторяя этот процесс, пока не останется ни одного числа. Тем самым получаем наши размеры гирь.

Олимпиада школьников «Надежда энергетики»

Вариант: 43991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ GE 28-39

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№3

~~разложим А на четыре части:~~

1	2	3	4
20zz	9yx	2019zy	1

поскольку $A4 = 1$, значит число не четное. Следовательно, делить $D = yx$ тоже нечётен. Значит, x - не четное число. Далее проверим ~~xy: A2 и A3 содержат~~ у число A без неизвестных нам знаков, а после переберём все возможные комбинации, подходящие под условия, такие как:

1) x - нечётное, не 5, т.к. не заканчивается на 0 или 5, значит $x = \{1, 3, 7, 9\}$

2) y - не 0, т.к. yx - двузначное

чтобы ~~делитель~~ 64 битного типа $A = 20009002019001$;

Перебираем $x_1 = \{1, 3, 7, 9\}$ {

Перебираем $y_1 = (0 \ldots 9]$ {

Перебираем $z_1 = [0 \ldots 9]$

{

чтобы 64б типа $A_1 = A + (x_1 * 10^2) + (x_1 * 10^7) + (x_1 * 10^{11})$;

$A_1 = A1 + (y_1 * 10) + (y_1 * 10^8) + (z_1 * 10^{19})$;

}

если $A1 / (x_1 + y_1 \cdot 10)$ в остатке дает 0 - выводим наши числа (корни подходят)

}

}

Асимптотика: $4 \cdot 9 \cdot 10$

№4

Олимпиада школьников «Надежда энергетики»

Вариант: 73991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ GE 23-39

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Чтобы выполнить задачу за минимальное время будем поддерживать ~~лог~~ ~~логическую перемещную~~, ~~одновременно~~, ~~встр~~ ~~ечались ли~~ ~~жёлтые камни~~.

~~логич. пер. Б - нет~~, Также создаем 2 ~~новых~~ линии, куда в которых наши камни и будет ответом.

Перебираем все камни по порядку:

1) Если камень тёмный ~~Б-да, иначе~~, вставляем его в вторую линию.
2) Если камень зеленый — перемещаем его в конец первой линии ~~массива~~ иначе

~~если Б-да, то~~ если камень красный — перекладываем его в начало

Создаем ~~список~~ (дек) куда мы можем добавлять элементы как в начало, так и в конец. Запишем туда нашу последовательность камней, изначально будем смотреть на первый.

~~Пока камень камня, на который мы смотрим, меньше N)~~

~~1) Если камень жёлтый, кол. жёлтых +1,~~
~~2) иначе если камень зеленый, кол. зел +1,~~
~~3) иначе кол. крас +1, если камень красный,~~ ~~кол~~крас +1,

Опять смотрим на первый камень.

Пока камень не последний, продолжаем:

1) Если камень желтый, смотрим на следующий иначе
2) Если камень красный, ставим его в начало, поскольку камни сдвинулись, продолжаем смотреть на тот же номер. Если ~~камень последний~~ красных ~~прерывается.~~ передвинули ~~больше~~ столько сколько ~~чем~~ количество красных, берем следующий номер.

Олимпиада школьников «Надежда энергетики»

Вариант: 73991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ GE 28-39

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

3) Если камень зелёный – ставим в конец, смотрим на тот же номер. Если номер последний – прерываемся. Если передвинули ~~больше~~ столько зеленых, ~~если~~ сколько их кол-во – прерываемся.
Тем самым мы получим отсортированный за асимптотику $2N$ (учитывая подсчет количеств)

№5

~~Сохраним все остатки от деления в массиве~~;

~~идем по массиву~~

~~1) Сохраняем сумму корней~~

если a, b и c имеют общий делитель – сокращаем их;

$\cancel{x_1 + x_2 = \frac{-b}{a}}$

$\cancel{x_1 * x_2 = \frac{c}{a}}$

доп ячейка $D = b^2 - 4ac$;

доп ячейка $x_1 = \frac{\sqrt{D} - b}{2a}$; $x_2 = \frac{-\sqrt{D} - b}{2a}$;

если x_1 – целое и ~~делится~~ на меньше m – выводим Да;

иначе если x_2 – целое и ~~делит~~ меньше m – выводим Да;

иначе выводим Нет,

Олимпиада школьников «Надежда энергетики»

МЭИ г. Москва	BS 88-49
Место проведения	шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 2 3101

ФАМИЛИЯ Волкова

ИМЯ Ия

ОТЧЕСТВО Владимировна

Дата рождения 22.04.2004 **Класс:** 10

Предмет Информатика **Этап:** Заключительный

Работа выполнена на 7 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

№ 1

Ответ: 2 датчика

Поставим датчики на 2 черты, показанные на рисунке (пронумеруем их 1 и 2), ~~то~~

Если загорелся (показал, что черта прочерчена) только первый датчик, то это этот знак.

Если только 2ой, то этот.

Если оба, то этот

Если ни одного, то этот.

В результате мы видим, что с помощью двух датчиков мы можем однозначно задать каждый из знаков.

Если мы добудем только 1 датчик, то он сможет задавать только два варианта (загорелся или нет), а у нас 4 знака, все из них одним датчиком задать невозможно.

+

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 83-49

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

N 3

a – массив, в котором указаны, какие числа записаны в ячейки таблицы (это и есть наша таблица)

```
N = int(input())
d = float(input())
s = float(input())
```

(*)

Вводим N, d и s. Если N не целое или d или s не числа (целые или дробные), программа выдаст ошибку.

(*)

```
not_find = True //это переменная, которая хранит значение типа bool (True если ещё не нашли нужный нам элемент и False если уже нашли)
```

Алгоритм

```
i = 0 //i обозначает текущий номер строки (нумерация с нуля)

while not_find and i<N: //выполняем, пока не найдем нужное значение или i станет равным N (тогда номер строки выйдет за границу массива)
    if i%2==0:
    //если номер строки чётный (с нумерацией с нуля)
        j = 0 // j обозначает номер столбца
        while not_find and j<N:
        //пока не нашли ответ или не вышли за границу массива
            if a[i][j] != "empty" and ... (**)
            //так мы обозначали пустые элементы
```

(**) продолжение на стр.3

```
(*) a = []
for i in range(N) //(выполняем N раз)
    ai = ~~list(map(int, input().split()))~~
    a.append(ai)
```

Считываем, разделяя по пробелам элементы новой строки

//Вводим N строк, в каждой N элементов через пробел. Если в этой ячейке нет числа, пишем вместо него строку "empty"

Доп. лист


```
(**)
and int(a[i][j]) ≥ d and
int(a[i][j]) ≤ s:
    //если число принадл. отрезку [d; s]
    ans_i = i+1
    ans_j = j+1
    //ans_i хранит номер строки ячейки, которую мы ищем (нумерация от 1)
    ans_j хранит номер столбца ячейки, нумерация от 1
    not_find = False
        //теперь мы нашли нужный элемент
  j += 1 //переходим к след. элементу строки
if i % 2 == 1: //номер строки нечет, нумер. с 0
    j = N-1 //номер столбца
    while not_find and j ≥ 0:
        //пока не нашли ответ или не дошли до самого первого элемента строки
        if a[i][j] != "empty" and
        int(a[i][j]) ≥ d and int(a[i][j]) ≤ s:
            ans_i = i+1
            ans_j = j+1
            //задаем значение аналогично выше
            not_find = False
        j += 1 //к след. эл. строки
i += 1 //переходим к след. строке
```

Доп. лист


```
if not not_empty: //не нашли элемент
    print("Такого элемента нет")
else:
    print(ans_i, ans_j) //выводим номер
        строки и столбца первой ячейки,
          нумерация с 1.
```

№5

```
a = int(input())
b = int(input())
c = int(input())
m = int(input())
//вводим a, b, c, m. Если это не целые
   числа, программа выдаст ошибку

if m <= 0:
    print("Неверный формат входных данных,
          m должно быть натуральным")
```

~~Если мы рассмотрим каждую цифру степени будем брать остатки от деления на m в степени будем брать остатки в сложении и сложении, вычитании, [illegible]~~

```
D = (b**2) - 4·a·c //считаем D, дискриминант
 if D < 0:
    print("Код ненадёжный") //если D < 0, решений нет
                 или sqrt(D) / корень из D
 else:
    x1 = ((-1)·b + √D)/2·a
    x2 = ((-1)·b - √D)/2·a
    if (x1 >= 0 and x1 < m) or (x2 >= 0 and x2 < 0):
        print("Код надёжный")
        //смотрим, чтобы решение лежало в множестве
остатков при делении нат. чисел на m, т.е. в множестве
  чисел от 0 до m-1 включительно
```

7

Доп. лист
Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS - 38 - 49

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

```
else:
    print ("Код ненадёжный")
    //какой код получится ненадёжным
```

№ 2

Посмотрим, как можно получить 40кг
Возьмём гири 27кг, 9кг, 1кг и 3кг
из 1кг и 3кг можно получить
1кг 3кг
$3-1=2$ кг (минус, если кладём на противоположную чашу весов)
$3+1=4$ кг
Из 1кг, 3кг и 9кг можно получить ~~все 1кг...4~~ все веса в промежутке 1кг...4кг.
$9-4, \ldots, 9$ кг ($9-4=5$) (Отнимаем т.е. кладём на противоположную из 9 все веса от 1 до 4)
$9, \ldots, 9+4$ ($9+4=13$ кг) (прибавляем все веса от 1 до 4)
то есть все от 1 до 13кг)
Если возьмём ещё 27 кг, сможем получить
все от 1 до 13кг
все от 27-13 до 27 кг ($27-13=14$ кг) (отнимаем все веса от 1 до 13)
все от 27кг до 27+13 ($27+13=40$ кг!), т.е все веса

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 7 3101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 88-49

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№4

1) $\frac{x^{2N-1}}{N} \cdot (-1)^{N-1}$

в Nом элементе ряда возводим число x в числителе в степень Nго по счёту нечётного натурального числа (1ый эл. – x^1
2ой – x^3
3ий – x^5
4ый x^7 и т.д.)

Nтое нечётное число = 2N − 1
(т.к. чётных и нечётных чисел одинаковое количество в списке нат. чисел от 1 до K если K чётное, если K нечётное, нечётных на 1 больше)

делим на N, т.к. первое число делим на 1, второе на 2 и т.д.

У нечётных (начинаем нумеровать с 1) чисел знак +, у чётных –. Чтобы получить это, возводим −1 в степень N − 1
(или N + 1, не важно, главное, что должна быть чётная степень (−1 в чётной степени = 1 в нечёт = −1) у нечётных N и наоборот)

2) ~~$\frac{S_{N+1}}{S_N} = -x^2 \cdot \frac{N}{N+1}$~~

~~$S_n = \frac{S_{N+1}}{-x^2 \cdot \frac{N}{N+1}} = \frac{S_{N+1}}{\frac{-x^2 N}{N+1}} = \frac{(S_{N+1})(N+1)}{-x^2 \cdot N}$~~ ...

~~Всего ... формулы~~

У нашей формулы из пункта 1) возведение в степень 2N-1 это 2N-1 умножений, возведение (−1)

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 88-49

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

в степень N-1 это (N-1) умножений, т.е.

Всего $2N-1+2+N-1$ арифм. операций

$= 2N+N-\cancel{1-1+2} = 3N$ арифм. операций

Теперь посмотрим на формулу из условия.

$$\frac{S_{N+1}}{S_N} = -x^2 \cdot \frac{N}{N+1}$$

$$S_{N+1} = \left(-x^2 \cdot \frac{N}{N+1}\right) S_N$$

$$S_{N+1} = (-1) \cdot x \cdot x \cdot \frac{N}{N+1} \cdot S_N$$

Всего 6 арифм. операций

Продолжение задачи N 2 (см. лист 5)

теперь с ~~помощью~~ гирь 1 кг, 3 кг, 9 кг и 27 кг. мы можем получить все веса от 1 кг до 40 кг.

Возьмём ещё гирю 50 кг

С помощью неё и всех предыдущих, прибавляя все веса от 1 кг до 40 кг, мы можем получить все веса от 50 кг до 90 кг,

а вычитая все веса от 1 кг до 40 кг, (кладя на другую чашу весов), от $50кг - 40кг = 10кг$ до 50 г

до 10 кг тоже можем получить

Ответ: 5 гирь 1 кг, 3 кг, 9 кг, 27 кг, 50 кг.

Олимпиада школьников «Надежда энергетики»

город Уфа

Место проведения

НЕ 23-21

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ ВОСКОВЩУК

ИМЯ ДМИТРИЙ

ОТЧЕСТВО МАКСИМОВИЧ

Дата рождения 19.04.2003 **Класс:** 10

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

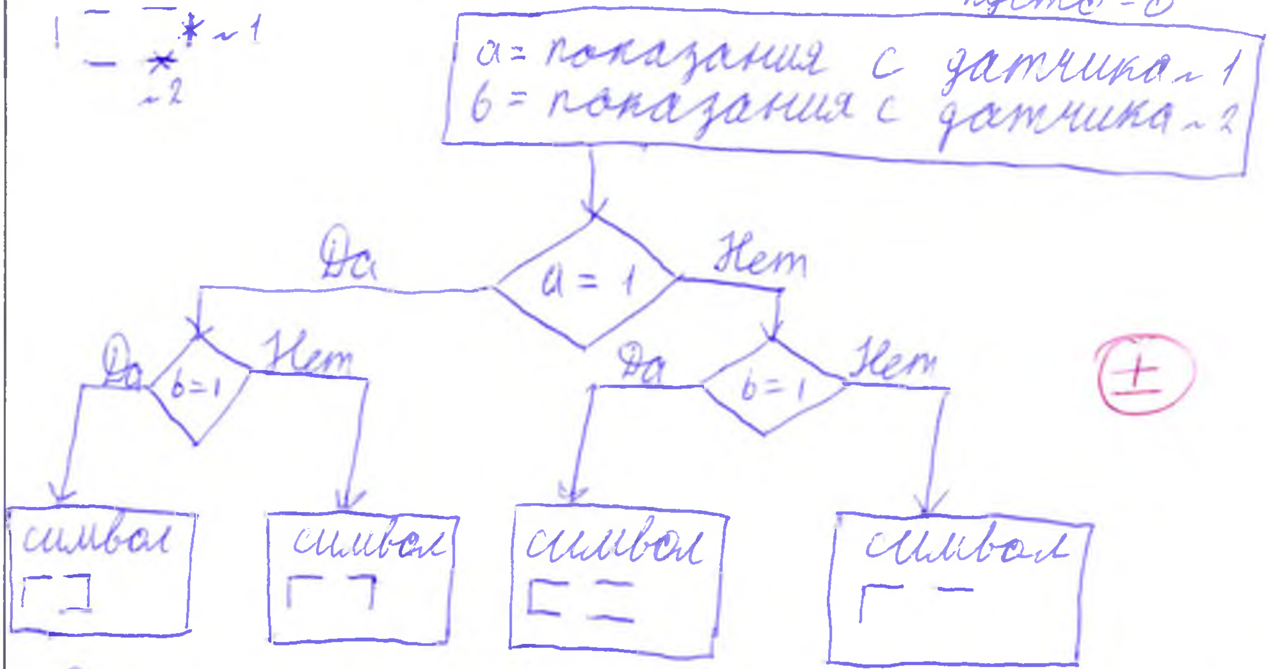
Работа выполнена на 3 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

1. Потребуется 2 датчика. Закрашено = 1
пусто = 0

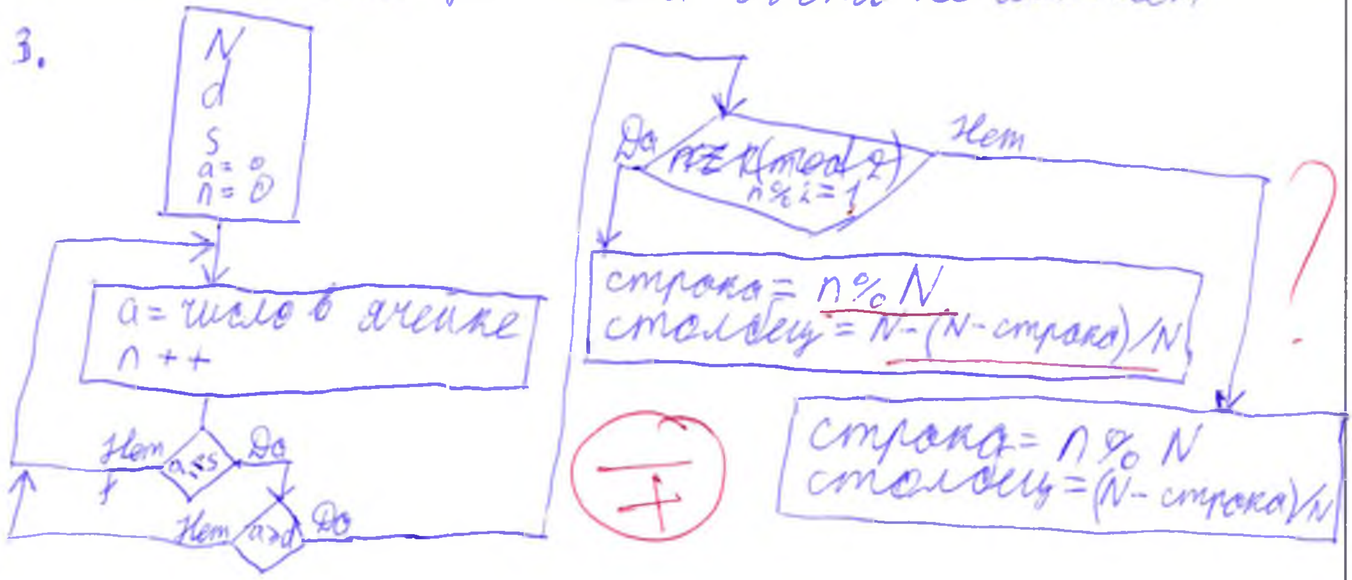


±

Доказательство, почему нельзя меньше. т.к. значения от датчиков получаются логические, то их вывод можно записать в виде двоичного кода, где датчик~1 – 1-ый разряд, а датчик~2 - 2-ой разряд и т.д.

Чтобы зашифровать 4 исхода потребуется 4 числа, что является максимальным числом в 2-ухразрядной системе. Значит 1-ого датчика быть не может

3.



?

∓

4. 1) ~~$(-1)^{N-1}$~~ $S_N = (-1)^{N-1} \cdot \frac{x^{2\cdot N-1}}{N}$

2) ~~Nea~~

операция	кол-во операций
$N-1$	1
$(-1)^{N-1}$	$N-2$
$2\cdot N$	1
$2\cdot N+1$	1
$x^{2\cdot N+1}$	$2N$ ~~ва~~
$\frac{x^{2\cdot N+1}}{N}$	1
$(-1)^{N-1}\cdot\frac{x^{2N+1}}{N}$	1
	$3N+3$

$(-1)^{N-1}\cdot\frac{x^{2N+1}}{N}$

операция	кол-во операций
x^2	1
$-x^2$	1
$N+1$	1
$\frac{N}{N+1}$	1
$-x^2\cdot\frac{N}{N+1}$	1
	5

+

$-x^2\cdot\frac{N}{N+1}$

чтобы считать этим способо нужно по очереди считать каждый элемент. Чтобы добраться от известного $N=4$ до N, нужно использовать формулу $N-4$ раз, т. е. $5(N-4)$ операций

$5N-20 \overset{?}{=} 3N+3$

$2N \overset{!}{=} 23$

При $N \geq 12$ второй способ выгоднее, иначе выгоднее первый

2. Понадобится 5 гирь с весами {81; 27; 9; 3; 1}
Двумя гирями можно ~~зшив~~ взвесть 9 значений, находящихся друг от друга на расстоянии n. Вес двух гирь должен быть n и 3n.
В диапазоне (1; 90) 87 элементов.

```
87 | 9
81 | 9
 6
```

Т. е. диапазон можно разбить на групп, ~~по~~ где в 9 будет 9 элементов и в одной только 6. Гирями {81; 27; 9} можно выставить любой вес, такой, что ⋮9 и входит в [0; 90]. А гирями {1; 3} можно выставить любой вес, у которого разница по модулю с весом, выставленным 3-мя гирями не больше 4. (т.е если $x \vdots 3$ & $x \in [0; 90]$, то можно выставить $[x-4; x+4]$)

+

5. Язык: C++

```
#include <iostream>
int a,b,c,m,i,j;
int main(){
cin>>a>>b>>c>>m;
if(a==0){cout<<"Ненадёжен";
        return 0;}
 for(i=0; i<m; i++)
    for(j=0; j<m; j++)
       if(i+j==(-1)*b/a)
          if(i*j==c/a){
            cout<<"Надёжен";
             return 0;}
 cout<<"Ненадёжен";
 return 0;
}
```

+

Олимпиада школьников «Надежда энергетики»

СОШ №20 город Новочебоксарск

Место проведения

QQ 84-47

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73991

ФАМИЛИЯ Григорьев

ИМЯ Роман

ОТЧЕСТВО Андреевич

Дата рождения 13.05.2004

Класс: 9

Предмет информатика

Этап: заключительный

Работа выполнена на 3 **листах**

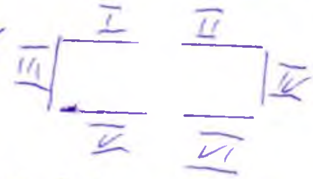
Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: Григ

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

№1

Чтобы определить кол-во датчиков, найдём мин. возможное кол-во определителей: датчики дают только сигналы да или нет, значит если у нас 3 знака, по двум ответам мы их не определим. Если датчиков 2, то один датчик ставим на IV позицию и проверяем на наличие деталь, если нет, то это 3 знак, а если есть проверяем еще и VI датчик, если от последнего наличие, то это 2 знак, а если нет то 1 знак. Таким образом если положительный сигнал на обоих дат., то это 2 знак, если только на IV — 1 знак, если только IV — 3 знак.



Ответ: 2 датчика.

±

№2

Так как максимальн~~ый~~ая ~~вес~~ масса груза 40кг, то для максимальной эффективности пусть сумма всех грузов будет = 40 кг. Чтобы ~~вз~~ взвесить 39кг, пусть I гиря будет массой 1кг, тогда просто уберем эту гирю из всех и весы уравняются. Для 38 просто I гирю переносим в другую чашу весов. Для 37 мы введем еще V гирю массой 3 кг и уберем её. По той же закономерности работаем дальше: убираем II гирю, убир. I гирю, добавляем на другую сторону I гирю, возвращаем I гирю на сторону весов без груза, а II добавляем к грузу.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

По этому принципу* доходим до груза с массой ~~3 кг~~ 31 кг. Добавляем III гирю массой 9 кг. По той же закономерности доходим до груза массой 13 кг, вводим последнюю гирю массой 27 кг: $1+3+9+27=40$ (кг).
По той же самой закономерности мы можем взвесить ~~все~~ любой груз массой от 1 до 40 кг.
Ответ: гири массой 1 кг, 3 кг, 9 кг и 27 кг.

+

*— принцип подробно описан в конце работы

№3

~~Для составления алгоритма~~

~~Д~~ Так как $\overline{yx}$ – число двузначное, то $y \neq 0$
Сам алгоритм:
~~для~~ $z=0$
Для ~~z=0~~; пока z однозначное число; (шагом +1,
для $y=1$, пока y однозначное число; (шагом +1,
±
для $x=0$; пока x однозначное число; (шагом +1,
Если $A \% \overline{yx} = 0$, то x, y и z – являются реш.
Таким образом мы перебераем все варианты x, y и z, а значит ничего не упустим.

№4

Так как мы не знаем и не можем найти цвета камней, но переберем все способы после того, как распределим цвета по III группам:

$\underbrace{I, I, \ldots I}_{I\ цвет}$ $\underbrace{II, \ldots, II}_{II\ цвет}$ $\underbrace{III, \ldots III}_{III\ цвет}$, Перебирать будем так: I II III, II III I, III I II, III II I, II I III, I III II.

Олимпиада школьников «Надежда энергетики»

Вариант: 73991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QQ 84-47

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

При таком способе перебора мы потратим 21 действий, т.к. каждый цвет мы перебираем по 2 раза.

Для того чтобы распределить цвета по группам воспользуемся алгоритмом.

Сравниваем I1 камень со всеми камнями, ~~если~~ если цвета совпадают перемещаем ~~его~~ этот камень и ставим перед I1.

Сравниваем цвета последнего камня со всеми оставшимися

если их цвета совпадают, перемещаем этот камень в конец.

№ 5

Вводим числа x и y — они будут определять числа в Z_m. Алгоритм:

Если в уравнении $ax^2+bx+c=0$, $b^2-4ac<0$, то этот код не надежный;

Иначе; если $\left(\frac{-b+\sqrt{b^2-4ac}}{2a}\right.$ или $\left.\frac{-b-\sqrt{b^2-4ac}}{2a}\right) = ((x+y)\%m$ или $(x\cdot y)\%m)$, то код надежный;

Иначе код не надежный.

№ 2

Для каждой гирьки мы имеем 3 состояния. Она на весах не на стороне груза, без ~~груза~~ гирьки, на весах со стороны груза. У каждого состояния разница в массу грузика. Кол-во целых масс ~~не считая~~ между состояниями m грузика − 1, то есть для 1 — это 0, а для 3 — 2, а 1 имеет 2 тон. сост

Олимпиада школьников «Надежда энергетики»

СФ МЭИ

Место проведения

НС 75-24

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ Грунтов

ИМЯ Дмитрий

ОТЧЕСТВО Игоревич

Дата рождения 27.01.2004

Класс: 10

Предмет информатика

Этап: Заключительный

Работа выполнена на 5 листах

Дата выполнения работы: 16 февраля 2020
(число, месяц, год)

Подпись участника олимпиады: Д

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Задание номер 1

Заметим, что в каждом символе 2 верхние и левая черты всегда присутствуют, значит нам не стоит проверять наличие этих черт => для распознания 1-го символа нам максимум понадобится 3 датчика. Также заметим, что для опознания символа не достаточно 1-го символа, так как символов 4, а состояний у 1 черты всего 2 (она есть / она отсутствует).

Так как одинаковых по написанию символов нет, то всех датчиков хватает.

Пронумеруем оставшиеся черты числами от 1 до 3. Таким образом 

Теперь попробуем выбрать пару черт так, чтобы можно было опознать со 100% вероятностью какой именно символ мы анализируем.

Возьмем пару (1,2), но в таком случае 1-ый и последний символы для нашей сис-мы будут одинаковы, т.к. на местах (1;2) в них отсутствуют черты => эта пара нам не подходит.

Тогда возьмем пару (1,3). Эта пара нам также не подходит, так как в первом и втором символах на этих позициях черты будут совпадать и наша сис-ма скажет, что это одинаковые символы.

Тогда попробуем взять пару (2,3). И тогда мы заметим, что расположив датчики в этих местах, мы сможем точно определить, какой символ у нас.

Ответ: для анализа 1 символа достаточно 2 датчика. +

Олимпиада школьников «Надежда энергетики»

Вариант: 73101 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ NC 75-24

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача №2.

Попробуем набрать максимальный W, имея всего 1 гирю. Логично, что если мы хотим набрать все веса от 1 до W имея всего 1 гирю, то мы должны будем взять гирю весом 1. Теперь попробуем взять гирю большего веса так, что бы максимизировать наше W. Логично, что нам будет не выгодно иметь 2 различных варианта собрать какой-то вес, из чего можем получить данное неравенство.

$V - W > W$, где V - вес искомой гири.

~~a W - максимальный~~

Тогда $V > 2 \cdot W$, но для того, что бы W выросло в значении, нам надо, что бы элемент $W+1$ присутствовал в нашей последовательности. Тогда $W+1 \leq V - W$.

Решая систему неравенств $\begin{cases} W+1 \leq V+W \\ V > 2 \cdot W \end{cases}$

мы получаем, что $V = 2W+1$.

+

Тогда по данной формуле и будем добирать веса гирь до того момента, пока $W < 90$.

В ответе мы получим 5 гирь с весами 1, 3, 9, 27, 81 кг.

Ответ: 5 гирь с весами 1, 3, 9, 27 и 81 кг.

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ NC 75-24

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задание №3.

Объявим матрицу размера $N \times N$

Инициализируем значения данной матрицы

Введем начало и конец отрезка $[d, S]$

Перебираем номер строки, по которой мы идем от 1 до N.

N 20 $d \le S$

{ Если (номер строки) mod 2 == 1, то

{ идем по нашей строке массива от 1-го элемента до N-гои проверяем.

если $d \le$ элемент массива $\le S$, значит

этот элемент лежит в заданном отрезке и мы нашли ответ ⇒

⇒ выводим номер столбца и номер строки и завершаем программу.

}

иначе, если (номер строки) mod 2 == 0, то

{ идем по строке массива от N до 1 и проверяем; если $d \le$ элемент массива $\le S$, то мы нашли ответ.

⇓

выводим номер строки и номер столбца завершаем програму.

}

}

Т.к. до этого шага программа не была завершена, то в нашем массиве нет элемента лежащего в отрезке $[d, S]$. Выводим, что победить невозможно и завершаем программу.

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ NC 75-24

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задание 4 $S(x) = x - \frac{x^3}{2} + \frac{x^5}{3} - \frac{x^7}{4} \dots$

Формула N-го слагаемого:

$$S_N = (-1)^{N+1} \cdot \frac{x^{(2N-1)}}{N}$$

Проверим её.

Найдем по этой формуле первое слагаемое ($N=1$)

$$(-1)^{1+1} \cdot \frac{x^{(2\cdot 1-1)}}{1} = 1 \cdot \frac{x}{1} = x$$

Найдем второе слагаемое ($N=2$)

$$(-1)^{2+1} \cdot \frac{x^{(2\cdot 2-1)}}{2} = -1 \cdot \frac{x^3}{2} = -\frac{x^3}{2}$$

Найдем третье слагаемое ($N=3$)

$$(-1)^{3+1} \cdot \frac{x^{(3\cdot 2-1)}}{3} = 1 \cdot \frac{x^5}{3} = \frac{x^5}{3}$$

Найдем четвертое слагаемое ($N=4$)

$$(-1)^{4+1} \cdot \frac{x^{(4\cdot 2-1)}}{4} = -1 \cdot \frac{x^7}{4} = -\frac{x^7}{4}$$

Для данных нам значений слагаемых проверка прошла успешно.

(1) $S_{N+1} = S_N \cdot \left(\frac{N}{N+1} \cdot (-x^2)\right)$

(2) $S_N = (-1)^{N+1} \cdot \frac{x^{2N-1}}{N}$

В формуле №1 мы используем значение предыдущего слагаемого, что обязует нас посчитать все слагаемые, которые идут до N, что мы не можем сказать про формулу №2. В формуле №2 для любого N, мы потратим одинаковое кол-во действий.

+

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇒ NC 45-24

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задание №5.

$x \in N$ Обозначим $x\%m$ за взятие числа

$x \ne < m$ x по модулю m

Введем значения переменных a, b, c, m
Переберем значения x от 0 до m-1 и проверим.

{ если $((a \cdot x \cdot x)\%m + (b \cdot x)\%m + c\%m)\%$
$\% m = 0$, то код является надежным и мы можем вывести Yes (обозначение, что код является надежным) и завершить программу.
}.

+

Если до этого момента программа не завершилась, то ответа для данного квадратного уравнения не существует в множестве Z_m и тогда наша программа говорит, что ответа нет.

Олимпиада школьников «Надежда энергетики»

СФ МЭИ

Место проведения

ИФ 84-93

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ Дементьева

ИМЯ Анастасия

ОТЧЕСТВО Алексеевна

Дата рождения 22.02.2002

Класс: 11

Предмет информатика

Этап: заключительный

Работа выполнена на 5 листах

Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

2. 1) Запишем все числа в массив a
2) отсортируем его

```
нач
 делать от i:=1 до n-1
  нач
   делать от j = i+1 до n
   нач
     если a[i] > a[j] то
     нач
       t := a[i];
       a[i] = a[j];
       a[j] := t.
     кц
   кц
  кц
кц
```

±

3) вывести элемент под номером k
(a[k]).

4. Ответ: 5 грузов (1, 3, 9, 27, 81)

+

Будем брать камни так, чтобы до добавления следующего мы могли собрать все числа
(т.е. например, мы можем сложить все числа от 1 до 5, значит следующее число таковым, что вычитав из него все числа от 1 до 5, мы дойдем до числа 5).
Возьмём 1 чтобы получить 2, нам надо 2+1=3. следующий груз 3
3=3; 4=3+1, следующий 5+3+1=9
Значит мы можем собрать все числа от 1 до 9+1+3=13, далее 14+13=27
Значит камень 27. Мы можем собрать все числа от 1 до 1+3+9+27=40.
41+40=81. Теперь мы можем собрать до 1+3+9+27+81=121

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ ИФ 87 - 93

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

3 1) Так как про длину брёвен (l) ничего не сказано, то есть 3 варианта ?
  1) длина ~~задаётся~~
  2) длина бревна = длина ущелья (L) − координата самого последнего бревна + 1
  3) высчитать все возможные варианты от 1 до 2) пункта.

2) Запишем все координаты начал брёвен в массив a.

3) Отсортируем его

(Пример сортировки

```
НАЧ
  ДЕЛАЙ от i = 1 до n-1
  НАЧ
    ДЕЛАЙ от j = i+1 до n
      ЕСЛИ a[i] > a[j] ТО
        t = a[i],
        a[i] = a[j].
        a[j] = t
    КЦ
  КЦ
```
)

4) Будем рассматривать координаты начала и конца всех брёвен, т.к между ними ничего не меняется

одно значение

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Возьмём переменную flag (целое)
i, k - которые обозначают координаты
S - кол-во бровей, max - максимальное кол-во бровей.

~~flag=1.~~ ~~S=0~~
ДЕЛАЙ от i=1 ДО n
НАЧ S=1. flag=1.
ДЕЛАЙ пока (i-k>0) и (flag≠0)
нач
Если a[i-k]+l ≥ a[i] то
нач
k=k+1;
S:=S+1;
кц
ИНАЧЕ
flag:=0;
кц
если max<S то
max:=S;
S:=1; flag=1;
Делай пока (i+k≤L) и (flag≠0)
нач
Если a[i]+l ≤ a[i+k] то
нач
k:=k+1;
S:=S+1
кц
иначе flag:=0;
кц
Если max<S то
max:=S.

к.ц.

5) вывести max

f

Олимпиада школьников «Надежда энергетики»

Вариант: 73111 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ ИФ 87-93

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

1. Ответ 3

Найдём отличительную черту у каждого символа.

тут такой нет

Поставим датчики на эти места (в последней выберем любой)

Если не загорится 1 датчик, то точно можно сказать, что это 4 символ,

Если не загорится 2 датчик, то точно можно сказать, что это 3 символ.

Если 3 датчик не загорится, то 1 символ.

Если загорятся все, то это 2 символ

Мы можем говорить точно, т.к. чёрточка, которая не загорается отсутствует только у одного из символов.

А все сразу присутствуют только у 2

-1

Олимпиада школьников «Надежда энергетики»

Вариант: 73111 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ ИФ 37-93

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

5. Создадим двумерный массив a[i,j]

i,j - координаты, на которой стоим.

n,j - координаты, которые запоминаем

```
нач
  ~~x := N~~ 1,
  x := N, i := N, j := 1, y := 1,
  делай пока (x > 0) или (y < N+1)
  нач
    делай пока (i ≤ N) и (j ≤ N)
    нач
      если a[i,j] > 0, то
      нач
        n := i;
        m := j;
      кц
      i := i+1;
      j := j+1;
    кц
    ~~x := x-1;~~ если x-1 ≤ 0, то
    ~~i := x~~ нач
    ~~j := 1~~   x := 1
                 y := y+1;
                 j := y;
               кц
               иначе
               нач
                 x := x-1;
                 j := 1;
                 i := x;
               кц
  кц
кц
```

Нужный ответ n, m

+

Олимпиада школьников «Надежда энергетики»

МБОУ „СОШ № 2" г.п.о. Бура-Хурхалы

Место проведения

НГ 34-80

← **Не заполнять**
Заполняется ответственным работником

шифр

Вариант № 73101

ФАМИЛИЯ ЕГОРОВ

ИМЯ МАКСИМ

ОТЧЕСТВО ЮРЬЕВИЧ

Дата рождения 01.07.2003 **Класс:** 10

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 6 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: ____________

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

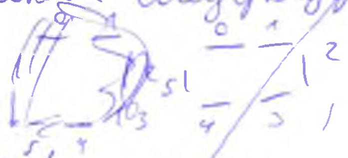
Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ NГ 34-80

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№ 1

Для каждого из знаков создадим массив 6 элементов (0 — ~~пять~~), где хранятся значения следующих частей прямоугольного шаблона:  значение 1 - наличие ребра, 0 - отсутствие, тогда алгоритм следующий:

нач

ответ = 8 // изначальное кол-во датчиков

знак 1, знак 2, знак 3, знак 4 - массивы для каждого прямоугольника

единиц // массив с позициями, где в каждом // прямоугольнике одно и то же значение

заполнение знак 1

заполнение знак 2

заполнение знак 3

заполнение знак 4

цикл от i = 0 до i = 5

Если знак1[i] == знак2[i] == знак3[i] == знак4[i], то

ответ = ответ - 1 // уменьшение ответа за // счет неиспользованного датчика

единиц.добавить(i) // i - неиспользованное положение

цикл от i = 2 до i = 2^6 // перебор двоичной маски

биты // массив позиций, где в маске стоит 1

можем_использовать = True // лог. переменная, показ. нет ли // в маске уже использованных // битов

max_eff // максимальная эффективность при данной ком.

заполнение битов, если бит уже в массиве единиц, можем ис = False

max_eff = 1

Если можем_использовать, то

цикл от j = 0 до j = 5

Если все биты в знак 1 == 1 или == 0 и все биты в знак 2 == 0 или == 1 и все биты в знак 3 == 0 или == 1 и все биты в знак 4 == 0 или == 1 и размер массива > 1)

max_eff = размер массива (битов)

~~ответ = ответ - (max_eff)~~

ответ = ответ - (max_eff - 1) // заменяем несколько // взаимозависимых // датчиков на 1

единиц.добавить(массив битов)

~~вывод(ответ)~~ / кон. цикл

~~кон алгоритма~~

цикл от i = 0, до i = 5, где

цикл от j = 0, до j = 5 если i ≠ j и

i не в единиц и j не в единиц и в каждом знаке значения i ≠ j, то

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

~~для ответа на вопрос из условия воспользуемся тем же алгоритмом:~~

~~1 мы можем не подрисовать ячейки 0, 1, 5, т.к. в центре значения они одинаковые~~

~~ответ = ответ − 1 (найдены две угаш... ячейки, ...)~~
~~log(ответ)~~

~~поп ам.~~

~~для ответа на вопрос из условия, воспользуемся тем же алгоритмом:~~

~~1 мы можем выбросить ячейки 0, 1, 5, т.к. в центре значения они одинаковые, ответ = 6 − 3~~

~~2 ищем ячейки, попадные в центре значения и не были использованы — это ячейки 2, т.к. она одна, то ответ не изменился~~

~~3 ищем пару взаимозаменяющих ячеек (пример: одна из них 1, другая 0, имеют в центре значение) таких в примере нет~~

~~Итоговый ответ: 3 (2, 3, 4 ячейки)~~

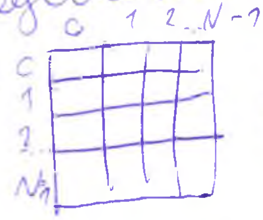
№2 легко заметить, что для взвешивания груза любого веса наиболее выгодно использовать гири, веса которых являются степенью 3, доходящих до веса, равного двум весам грузов, т.к. груз весом $\frac{3^x-1}{2}$ можно взвесить ровно x гирь (пример из условия W = 40, гири 1, 3, 9, 27), тогда, чтобы взвесить груз до W = 90, нужно 5 гирь → 1, 3, 9, 27, 81 +

Доказать это можно тем фактом, что

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

любое число N можно разложить на степени тройки, добавив к нему не более $\lceil \log_3(2N) \rceil - 1$

№3 Представим таблицу следующим образом:



двумерный массив, где первый индекс - строка, второй - столбец

±

нач ~~вывод~~ таблицы d s //диапазон значений)
~~ввод~~ ввод(N,d,s)
num-массив [N][N]
заполнение массива
pos=0 // направление движения
dx[2]={1,-1} ← движение из ~~...~~ столбцам
~~...~~
y=0 - строка
x=0 - столбец

N>0?

пока (y<N)
подход_pos={} //массив позиций, где элементы с [d;s]
пока (x<N и x>-1)
Если num[y][x] ≥ d и num[y][x] ≤ S, то
подход_pos.добавить(пара(y,x))
x = x + dx[pos]
pos=(pos+1)%2
x = x + dx[pos]
y = y+1
Если массив подход-pos не пуст, то
Если „по ходу движения" - движение ⇒
ответ = подход_pos[0]
иначе ⇐
ответ = подход_pos[~~размер~~ размер(подход-pos)-1]
~~вывод(ответ[0], ответ[1])~~ //ответ[0] подходит y, //ответ[1] подход x, //нумерация с 1
~~скончание алгоритма~~
~~вывод(Такое число не найдено)~~
вывод(ответ[0]+1, ответ[1]+1) // ответ[0] - //строка //ответ[1] - //столбец //нумерация с 1
завершение программы

вывод("подходящий элемент не найден")

№4 Рассмотрим первые числа последовательности детально

N	1	2	3	4
ч	x	x^3	x^5	x^7
з	1	−2	3	−4

, тогда

1) N-ный член последовательности равен

$$P/N = \frac{x^{(N\cdot 2-1)}}{N\cdot(-1)^{(N+1)\%2}},$$

2) При реализации 1 варианта можно получить 2 результата
① - долгое возведение в степень → $(N\cdot 2-1)$ - умножений, 2 деления
② - быстрое возведение в степень $\log_2(N)$ + 2 деления + накладные расходы

2 вариант -
~~4 деления~~ 2 умножения 3 деления

в ① 1 вариант выгоднее при $N \le 34$, при $N > 4$ выгоднее 2
во (2) 1 вариант выгоднее при $N < 16$
при $N \ge 16$ выгоднее 2

№5. Существует 8 вариантов содержимого a, b, c

a	b	c
0	0	0
0	0	не0
0	не0	0
0	не0	не0

a	b	c
не0	0	0
не0	0	не0
не0	не0	0
не0	не0	не0

т.к. множество Z_m содержит остатки от деления на числа на m, то все числа в Z_m - неотрицательные, принадлежат $[0; m-1]$, алгоритм вычисления следующий

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ NГ 34-80

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

```
Нач
 a, b, c, m // коэффициенты уравнения и услов. Ингпонть
 Ввод (a, b, c, m)
 Если (a == 0 и b == 0 и c == 0), то
     Вывод ("да, содержится") // имеет корень в (-∞; +∞)
 иначе если (a == 0 и b == 0 и c ≠ 0), то
     вывод ("нет, не содержится") // у уравнения нет корней
  иначе если (a == 0 и b ≠ 0 и c == 0), то // корень уравнения → 0,
     вывод ("да, содержится")                //  m > 0
  иначе если (a == 0 и b ≠ 0 и c ≠ 0), то
    если -c / b < m и -c / b ≥ 0, то
       вывод ("да, содержится")
    иначе
       вывод ("нет, не содержится")
 иначе если (a ≠ 0 и b == 0 и c == 0), то // корень ур. → 0, m > 0
     вывод ("да, содержится")
 иначе если (a ≠ 0 и b == 0 и c ≠ 0), то
      если -c / a ≥ 0, то
           если sqrt(-c / a) < m, то
             вывод ("да, содержится")
           иначе
             вывод ("нет, не содержится")
      иначе
        вывод ("нет, не содержится")
 иначе если (a ≠ 0 и b ≠ 0 и c == 0), то // корень → 0, m > 0
     вывод ("да, содержится")
 иначе
     d = b · b - 4ac
     если d ≥ 0
        x1 = (-b + sqrt(d)) / (2a)
          x2 = (-b - sqrt(d)) / (2a)
        если ((x1 ≥ 0 и x1 < m) или (x2 ≥ 0 и x2 < m)), то
            вывод ("да, содержится")
        иначе вывод ("нет, не содержится")
     иначе вывод ("нет, не содержится") // решение в комплексных числах
```

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1 Представим принадлежащими в сечении массива, где ячейки помещены обозначаются следующее: $\circ\,|\,\overset{1}{\_}\ \overset{2}{\_}\,|^{3}$, $\underset{5}{\_}\ \underset{4}{\_}$ чтобы найти необходимое минимальное количество датчиков воспользуемся битовыми масками, где 0 – наличие датчика в выключенном положении, 1 – наличие датчика в активном положении, 2 – отсутствие датчика, перебираем из 2.

Очевидно, что в положение (2) сразу можно перевести ячейки, равные в каждой из четырех знаков: в нашем случае получится следующая таблица

	0	1	2	3	4	5
1	2	2	2	1	0	0
2	2	2	2	1	1	0
3	2	2	2	0	1	1
4	2	2	2	0	0	0

, можем заметить, что убрав датчик с позиции 5 мы все еще можем однозначно установить изгиб => ответ в нашем примере → 2 датчика.

В общем же случае мы будем перебирать маску, пока положение, где не будет равных в 4-х клеточках масок, ответом же будет маска с наибольшим количеством (2)-отсутствующих датчиков.

Асимптотика в худшем случае N^2, где N – кол-во черт в знаке

Ответ: 2 датчика.

Олимпиада школьников «Надежда энергетики»

СОШ 20 г. Новочебоксарск

Место проведения

КМ 34-76

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73 111

ФАМИЛИЯ КАПРАНОВ

ИМЯ СТЕПАН

ОТЧЕСТВО СЕРГЕЕВИЧ

Дата рождения 02. 02. 2002

Класс: 11

Предмет информатика

Этап: Заключительный

Работа выполнена на 5 **листах**

Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: Капр

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 73 III

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ КМ 34-76

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1

1) Заметим, что левая черта имеется у всех знаков, а правая не имеется у одного, значит мы всегда сможем опр. информацию, если установить датчик на правую черту

2) Если мы посмотрим на 3 элемента, у которых правая черта имеется:

|¯¯| |¯¯] |__| , то заметим, что у двух из них сверху 2 черты, а у третьего - ни одной. Поставив датчик например на правую верхнюю черту, мы сможем различить 3й элемент всегда.

3) У оставшихся двух элементов: |¯¯| , |¯¯|- имеется одно различие правая нижняя черта. Установив туда датчик, мы сможем однозначно определить любой из 4-х знаков, т.е необходимо 3 датчика.

Чтобы доказать, что 3 минимальное кол-во, докажем, что не возможно за 2 датчика определить знак однозначно.

Для того, чтобы 4 знака однозначно определялись 2-мя датчиками, нужно потребовать выдачи следующих сигналов

{есть {нет {нет {есть
{нет {есть {нет {есть - здесь каждый датчик определяет есть черта или нет. Т.е нам нужно найти два таких место, чтобы у двух из 4-х знаков там была черта, а у двух других там их не было.

Из рассуждений 1-3 ясно, что это не может быть ни боковая, ни верхняя черта. При этом левая нижняя имеется у 3-х знаков. Значит, данный набор не возможно однозначно определить 2-мя датчиками
ч.т.д

Ответ: 3 датчика.

Лист 01 из 05

Олимпиада школьников «Надежда энергетики»

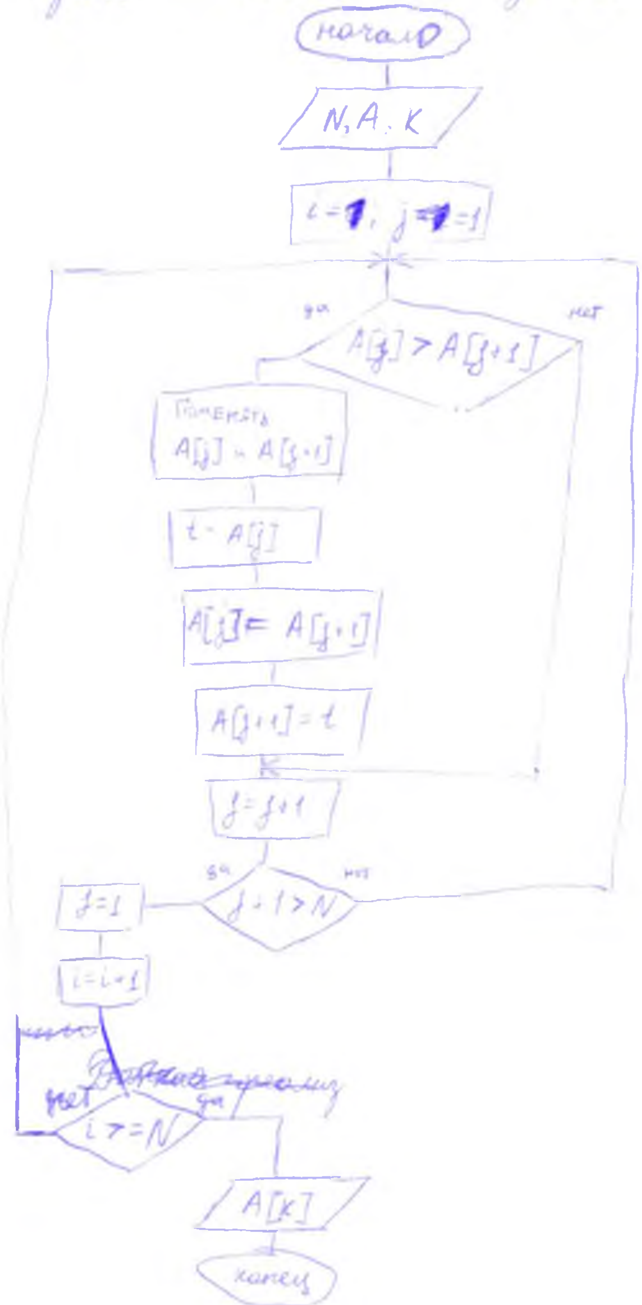
Вариант: 47 III

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ КМ 34-76

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№2 Отсортируем колонку за $O(N^2)$ и выведем k-ый элемент за $O(1)$. Общее число операций при этом будет $O(N^2)$

Пусть A - массив чисел длиной N



Сортировка пузырьком работает за $O(N^2)$. (Внутренний цикл заработает за N раз, Внешний цикл за N раз, $N \cdot N = N^2$)

Вывод происходит за $O(1)$.

Неоптимальная реализация пузырька

F

Олимпиада школьников «Надежда энергетики»

Вариант: 47111 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ КМ 34-76

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

15

Пусть нумерация строк идет сверху вниз от 1 до N, нумерация столбцов слева направо от 1 до N. Начинаем в ячейке [1, N].

Будем хранить последнее положительное число в виде его положения в этой таблице. Пусть X, y - номер строки и столбца этого числа.

Всю таблицу можно пройти за $O(n^2)$, при этом внешний цикл будет работать по разному в первой половине таблице и во второй.

Внутренний цикл будет выполняться следующим образом:

1) ~~Перейти из~~ Если хотя бы одна из i или j равна N - выйти из цикла.

2) Перейти в ячейку [i+1; j+1, т.е i=i+1; j=j+1;

3) Если A[i ; j] > 0, то X=i ; y=j ;

+

Теперь нужно определить работу внешнего цикла т.к внутренний всегда делает одно и то же. Посмотрим, где начинает, и заканчивает работу внешний цикл:

начало	[1, N]	[1 ; N-1]	[1 ; N-2]	...	[1 , 1]	[2 ; 1]	[3 ; 1]	...	[N ; 1]
конец	[1; N]	[2 ; N]	[3 ; N]		[N; N]	[N; N-1]	[N; N-2]		[N ; 1]

Внешний цикл меняет начальное положение по одному принципу первую половину (до главной диагонали) и по другому принципу ~~после~~ во второй половине. Введём переменную-флаг, принимающую либо 1, либо 0 (f=1 или f=0). Изначально f=1. Как только мы выходим из внешнего цикла, смотрим на значение f. Если f=1, то начальное положение ставим так j = N-i; i = 1; ~~если~~ Далее, если j=1 и i=1, то меняем значение f = 0. На следующих проходах цикла если f=0, то меняем начальное положение так: i = N-j; j=1. Таким образом, мы пройдём все ячейки таблицы. Наконец, когда после прохода внутреннего цикла передаётся положение [N, 1] т.е i=N. j=1, мы выходим из цикла и выводим значение A[X, y]

№4

Пусть существует какой-то набор гирь, состоящий из 4-х штук, при этом они могут взвесить любой груз от 1 до 40 кг. Заметим, что мы можем решить аналогичную задачу для $Q=120$ кг, если добавим гирю 80 кг:

1) Если груз $x \in [1;40]$, то мы можем его взвесить изначальным набором

2) Если груз $x \in [41;80]$, то мы можем дополнить его до 80 изначальным набором и сравнить с гирей 80 кг.

3) Если груз $x \in [81;120]$, то мы можем его составить из изначального набора и гири 80 кг $(80+40=120)$

Такие рассуждения можно использовать, чтобы определить остальные грузы: $\frac{80}{120} = \frac{2}{3} \Rightarrow$ пусть гиря в изначальном наборе имела вес $\frac{2}{3} \cdot 40\text{кг} = \frac{80}{3}\text{кг} = 26\frac{2}{3}\text{кг} \approx 27$ кг. Если одна из изначальных 4х гирь имела массу 27 кг, то остальными тремя можно было измерить $40-27=13$кг. Значит из набора 3х гирь, максимальная должна иметь массу $\frac{2}{3} \cdot 13\text{кг} \approx 9\text{кг} \Rightarrow$ $\Rightarrow$ последними двумя можно было измерять массы $13-9=4\text{кг} \Rightarrow 4 \cdot \frac{2}{3} = \frac{8}{3} \approx 3 \Rightarrow$ $\Rightarrow$ набор гирь: 1, 3, 9, 13, 27. Этим набором можно взвесить любой груз Q от 1 до 40 кг. Значит, набором 1, 3, 9, 13, 27, 80 можно взвесить любое Q от 1 до 120 кг исходя из рассуждений выше. Данный набор является наименьшим по кол-ву т.к. набор из 4х гирь в таком случае не сможет содержать гирю массой 1кг, без которой невозможно измерять $Q=1$кг. Значит набор наименьший.

Ответ: 1, 3, 9, 13, 27, 80

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№3

Т.к мы не знаем, в каком именно направлении лежат брёвна, будем считать, что можно принять координаты начала брёвен за окружности с одинаковыми радиусами. Чтобы найти самое большое скопление брёвен, найдём, при какой минимальной длине брёвен имеется одно наибольшее скопление (Если при меньшей длине таких скоплений стало два или более ~~дракон~~ не сможет определить, где находятся богатыри). Для поиска минимально возможной длины брёвен воспользуемся бинарным поиском. Пусть изначальная предполагаемая длина брёвен равна длине ущелья. Находим наибольшее скопление. Теперь делим длину брёвен пополам, если все ещё имеется одно скопление, продолжаем сужать границы, ~~деля на~~ находя средне арифметическое границ. Если появилось несколько скоплений, меняем левую границу на среднее арифметическое текущих границ. Таким образом, можем определить возможное место нахождения богатырей с любой точностью, если поставим условие на размеры границ рассматриваемого участка (Если радиус $\in [a; b]$ таких, что $\frac{a+b}{2} < \Delta X$, где ΔX - желаемая точность).

Для того, чтобы посчитать величину скопления в одной точке, нужно просто подставить её в неравенство $(X-x_i)^2 + (y-y_i)^2 \leq R^2$, где R - текущий радиус бревна $(x_i; y_i)$ - координаты начала брёвен, где $i \in [1 \dots N]$, $i \in \mathbb{N}$ Такую операцию можно проделать для любого кол-ва точек. Остаётся на каждом шаге предполагаемого радиуса подсчитать максимальное кол-во брёвен, пересекающихся в одной точке. Получим предполагаемую область, где находятся богатыри с любой точностью.

Олимпиада школьников «Надежда энергетики»

МЭИ, г. „Москва"

Место проведения

НЯ 67-30

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ Кирь

ИМЯ Валентин

ОТЧЕСТВО Александрович

Дата рождения 24. 01. 2002.

Класс: 11

Предмет информатика

Этап: Заключительный

Работа выполнена на ______ **листах**

Дата выполнения работы: 16.02.2020
(число, месяц, год)

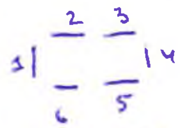
Подпись участника олимпиады: ____________

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

№ 1

Давайте для канала пронумеруем каждую черту по часовой стрелке, начиная с самой левой:

Представим, что у нас есть все датчики и запишем в таблицу номера черточек, которые существуют у того или иного символа (1-существует, 0-нет):



Символ \ Черточка	1	2	3	4	5	6
1	1	1	1	1	0	0
2	1	1	1	1	1	0
3	1	1	1	0	1	1
4	1	0	0	1	1	1

Запишем свои наблюдения:
- Можно исключить 1-ю черту вовсе, т.к. у всех символов значение идентично.
- можно исключить 2-ю черту, т.к. её значение идентично 3-ей.

У каждого датчика есть значение 0 и значение 1, соответственно, мы можем найти максимально возможное кол-во идентификационных булевых значений с „идеальными символами" - это будет первая степень двойки, не большая данного значения, для 4 - это 2, то есть идеальные символы должны иметь такие две черты, что в таблице они бы записались следующим образом

Символы \ Значения черточек	1	2
a_1	x	x
a_2	x	y
a_3	y	x
a_4	y	y

где x, y могут равняться 0 и 1, при этом $x \neq y$, а a_1, a_2, a_3, a_4 - номера символов от 1 до 4 в разных вариациях.

В данных нам символах есть черта № 6, которая удовлетворяет этому условию, но второй необходимой черты нет, следовательно 2 датчика использовать точно не получится

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ 67-30

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

таким образом черта 6 разделяет символы на две группы: 1, 2 и 3, 4, соответственно

далее нам необходимо найти ~~…~~ те черты, которые помогут нам отличить 1 от 2 и 3 от 4: для того, чтобы отличить 1 от 2 нам подходит только 5-ая ~~…~~ черта: если она есть, то перед нами ~~…~~ 2-й, если нет – 1-й символ (при условии, что ~~…~~ 6-ой ~~…~~ черты нет). Далее, нам необходимо отыскать черту, с помощью которой можно будет отличить 3-й символ от 4-го: для этого подойдёт как 3-я, так и 4-я черта, мы выберем 4-ю.

Таким образом с помощью ~~…~~ 3-х датчиков можно отличить 4 символа, занесем данные в таблицу

символы	4-я	5-я	6-я
1	1	0	0
2	1	1	0
3	0	1	1
4	1	1	1

(показания на датчиках; номера черт)

можем заметить, что показания каждого символа отличаются, что.

+

Ответ: 3 черты, не меньше (датчика)

№2

Буду писать алгоритм на псевдокоде в стиле python, т.к. по моему мнению, он сильно похож на нормальную речь

```
a[n] # колонка чисел, где под a[i] находится i-й элемент,
K # ~~...~~ искомый номер числа K∈[1;n]   i∈[0;n), i∈Z
for i in range(n): # переберём все i от 0 до n ([0;n))
    for j in range(i+1, n): # переберём для каждого i j от i+1 до n
        if a[i] > a[j]: # если элемент a_i (э-т с меньшим номером) больше a_j (э-т с большим), то меняем их местами
            save = a[i] # запомнили a[i]
            a[i] = a[j] # заменили a[i] на a[j]
            a[j] = save # заменили a[j] на a[i] из буфера
```

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-30

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Таким образом, у нас получился отсортированный массив, состоящий из элементов данной книзу колонки

print a[k-1] # Выводим ответ на задачу.

Вывели (k-1)-й элемент массива, т.к. номера колонки начинаются с 1, а номера массива с 0.

ответ: a[k-1], а кол-во операций $\frac{N \cdot (N+1)}{2} < N^2$ (+)

№ 4.

Пойдём от меньшего к большему:

пусть X_i - множество ~~~~ весов, которые мы можем получить с помощью i ~~~~ гирь разного веса, а A_i - множество весов этих самых гирь:

Для начала получим веса с помощью 1-ой гири: чтобы веса не имели „дыр" и мы могли каждый раз, добавляя новую гирю, увеличивать веса поступенно мы начнем с гири массой 1. Т.о. $X_1: [0; 1], A_1: \{1\}$

Далее возьмём гирю на (X_1 max (наибольшую массу) $\cdot 2+1$), чтобы положив максимум из того что можем получить с прошлым кол-вом на левую чашу, а новую гирю на правую, мы получили массу, следующую после максимальной. Т.О. $X_2: [0; X_{1max} \cdot 2+1+X_1 max]$; $X_2: [0; 4]$

Т.К. 2 мы можем получить 3 - 1, 3: 3, 4: 4+1. $A_2: \{1; 3\}$

Далее следуем тому же принципу

$X_3: [0; 13], A_3: \{1; 3; 9\}$

$X_4: [0; 40], A_4: \{1; 3; 9; 27\}$

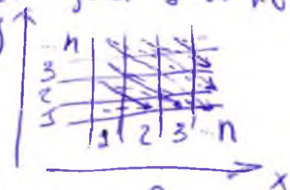
на четырех гирях мы уже можем получить $\forall$ вес при Q=40. Делаем следующую итерацию:

$X_5: [0; 121], A_5: \{1; 3; 9; 27; 81\}$, при 5-ти мы уже можем работать с Q = 120, что и требовалось найти.

Ответ: минимальное число гирь: 5, гири: {1; 3; 9; 27; 81}

№ 5

Т.К. нумерацию таблицы нам не дали, я пронумерую её самостоятельно: пусть ось X будет слева → направо (самая левая колонка будет 1-ой по оси x, а самая правая – n-ой), а ось Y – снизу ↑ вверх (самая нижняя строка будет 1-ой по Oy, самая верхняя – n-ой)



пусть координатой по Ox будет cond.x, а координатой по Oy – cond.y, а номера в таблице будут храниться в массиве a[n+1][n+1].

Т.О. напишем код в стиле как во 2-ом задании

```
While cond.x > 1 and cond.y > 1:
    print a[pos.x][pos.y] # выводим число данное в таблице
    if pos.y == 1:
        pos.y = pos.x - 1
        pos.x = 1
        continue # переходим на сл. итерацию цикла
    if pos.x == n
        pos.x = pos.y - 1
        pos.y = n
        continue # переходим на сл. итерацию цикла
    pos.x = pos.x + 1
    pos.y = pos.y - 1
print(a[1][1] # выводим последнее число в таблице – координаты выхода
```

таким образом, мы идём по таблице так, как сказано в условии и выводим значения в этой самой таблице.

№ 3

~~предположим, что я понял условия правильно: координаты начала бревен - это ... координаты левой и правой границы на которых расположились бревна, l - длина одного бревна, n - всего бревен, и бревна падают в свободное пространство (каждое последующее~~

pair < int , int > b[n] — массив координат расположения бревен

b.f - по высоте   b.s - по длине (по оси X, если так удобнее)

~~сделаем сортировку:~~

пройдёмся по всему массиву и запишем в переменную pair<int, int> max координаты самого высокого древка. т.к. самое высокое скопление существует, то это самое высокое бревно тоже должно существовать.

```
for i in range(n):
    if mx.f < b[i].f:
        mx.f = b[i].f
        mx.s = b[i].s
```

~~print ...~~

print "[", mx.s, ";", mx.s + l, "]", где

l - длина бревна. т.о. мы выведем диапазон координат, на которых могут находиться наши богатыри.

Олимпиада школьников «Надежда энергетики»

МЭИ, г. Москва

Место проведения

Мя 67-13

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ КУЩЕНКОВ

ИМЯ МАКСИМ

ОТЧЕСТВО ВАЛЕРИЕВИЧ

Дата рождения 17.09.2002

Класс: 11

Предмет ИНФОРМАТИКА

Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 5 **листах**

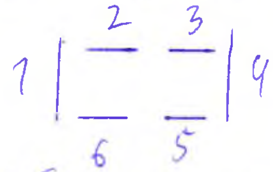
Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

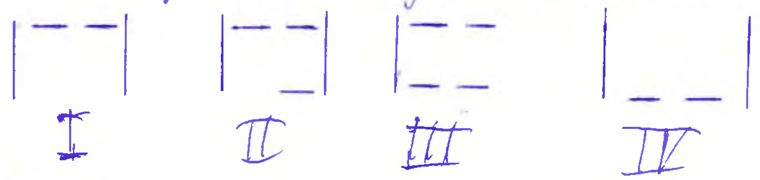
Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

№1.

Обозначим каждую черту прямоугольного шаблона символом:



Рассмотрим и обозначим символами знаки:



Заметим, что только у символа III непрочерчена черта 4, а черта 1 прочерчена у всех символов ⇒ определение состояния черты 1 не позволит определить, какому знаку она принадлежит; определение символа 4 позволит однозначно определить знак III.

Заметим, что группы символов 2 и 3 совпадают у знаков I и III, а группы 5 и 6 совпадают у знаков III и IV. ⇒ для определения знака III необходимо установить датчик на черту 4.

Рассмотрим знаки I, II и III: у знака II непрочерчена черта 6, у знака I непрочерчены черты 5 и 6, у знака IV черты 5 и 6 прочерчены. Для однозначного определения знака II необходимо установить датчики на обе черты (5 и 6). Таким образом, будет возможно определить и знаки I и IV.

Рассмотрим знаки I, II, III, IV, составленные только из черт 4, 5, 6. Обозначим непрочерченную черту как ~

I ~ ~ | II ~ — | III — — } IV — — | ⇒ Каждый знак можно однозначно определить с 3 датчиками

±

№2.

Обозначим колонку длины N как a, пронумеруем индексы колонки от 1 до N.

1	2	3	4	5	6	7 N

В ячейках записаны целые числа. Необходимо найти k-ое по возр. число в массиве. Чтобы найти это число, необходимо, начиная с числа k проверять, какое оно по возрастанию среди чисел, записанных до него (до числа с индексом k нет необходимости проверять, какое оно по возрастанию, т.к. k-ым по возрастанию оно точно не будет).

Алгоритм поиска (имея N, k; a - обозначение массива чисел):

```
нц для i от k до N включительно:
    c=0
    m=a[i]
    нц для j от i до 1 включительно:
        если a[j] < m то:
            m = a[j]
            c = c+1
    кц
кц
если c=k то: вывод a[i].
```

№3.

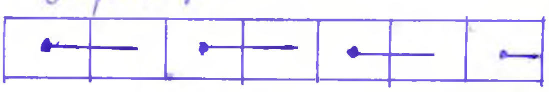
Обозначим ущелье как массив чисел (нулей) длиной N (I голова определила длину ущелья).

0	0	0	0	0	0	0
0	1	2	3	4	5	6

I голова определила координаты начала брёвен, II голова определила, что их длина одинаковая и число их k штук.

Обозначим координаты начал брёвен как точки (для примерочных длину отметим как прямые, выходящие из точек).

Пример:



1	1	1	1	1	1	1

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ 07-13

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Заметим, что конец бревна может выходить за пределы ущелья. Из предположения третьей головы следует, что при опр. длине брёвен k существует такой элемент массива, который будет наибольшим. По ходу выполнения алгоритма будем увеличивать k на 1, если максимальных элементов несколько.

Запишем алгоритм (имея массивы a (из нулей) и b (1 или 0 в зависимости, является ли ячейка координатой начала бревна), длину ущелья N, число брёвен n):

```
~~нц для~~  ~~k = 1, k:1 a = массив нулей~~   k=1
                                         2: a=массив нулей
нц для i от 0 до N невключительно:
    если b[i] = 1 то:
        нц для j от 0 до k невключительно:
            ~~a[i+j] = a[i+j]+1~~
            если i+j < N то:
                a[i+j] = a[i+j]+1.
кц      кц
m = a[0]
ans=0
c = 1
нц для i от 1 до N невключительно:
    ~~если a[i] > m~~
    если a[i] = m то: c = c+1
    иначе: если a[i] > m:
              m = a[i]
              ans = i
кц
если c=1 то: вывод ans, конец
иначе:
    k=k+1
    возврат к строке 2.
```

Алгоритм a = массив нулей: нц для i от 0 до N невключительно:
a[i] = 0

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ 67-13

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№4.

Известно, что любое число можно записать в двоичной системе счисления.

Будем брать гири весом, равным степеням двойки, рассмотрим, какое максимальное число можно суммированием получить: $1 \to 1$, $1+2 \to 3$, $1+2+4 \to 7$, $1+2+4+8 \to 15$, $1+2+4+8+16 \to 31$, $1+2+4+8+16+32 \to 63$

Заметим, что при наличии n гирь с весом, равными последовательным степеням двойки, можно получить число 2^n-1

Имея 6 гирь, можно взвесить груз в 63 кг =>

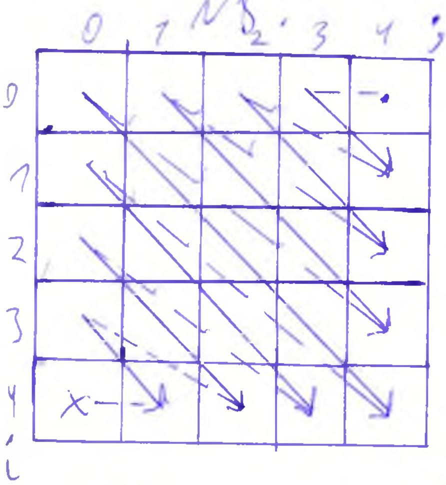
=> для взвешивания груза с $Q = 120$ кг потребуется больше 6 гирь.

Имея 7 гирь, можно взвесить груз в $2^7-1=127$ кг =>

=> 7 гирь достаточно для взвешивания груза с $Q=120$ кг.

Ответ: 7 гирь весом 1, 2, 4, 8, 16, 32, 64 кг соответственно.

№5.



Пусть a – квадратная таблица размером N·N, в ячейках которой записаны целые числа.

Так как движение по таблице возможно только в соответствии со схемой => необходимо последовательно проверить все ячейки маршрута, записывая номера строки и столбца последнего положительного числа (перезаписывать, если координаты положительного числа, встреченного по маршруту). Выпишем индексы ячеек в порядке прохождения по ним в примере при $N=5$:

(0;3), (1;4), (0;2), (1;3), (2;4), (0;1), (1;2), (2;3), (3;4), (0;0), (1;1), (2;2), (3;3), (4;4), (1;0), (2;1), (3;2), (4;3), (2;0), (3;1), (4;2), (3;0), (4;1),

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ VI-13

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Заметим, что до прохождения по главной диагонали (от (0;0) до (4;4)), начиная с (0;3), индексы ячеек увеличиваются ~~до~~ на 1, пока один из них не станет равен 4; тогда индекс столбца уменьшается на 1 ((0;2)). После прохождения по главной диагонали аналогичным образом от (1;0) индекс строки увеличивается до (3;0).

Алгоритм (имеем таблицу а размерами N·N):

```
pol = 0
~~НЦ~~ ans_i = 0
ans_j = 0
НЦ для j от N-2 до 0 включительно:
|   i = 0
|   если a[i;j] > 0 то: pol = a[i;j]
|                       ans_i = i
|                       ans_j = j
|   os = 1
|   НЦ пока i+os < N и j+os < N
|   |   если a[i+os;j+os] > 0 то: pol = a[i;j]
|   |                             ans_i = i+os
|   |                             ans_j = j+os
|   |   ~~pol = a[i;j]; ans_i = i+os; ans_j = j+os~~
|   |   os = os+1
|   КЦ
КЦ
НЦ для i от 1 до N-1 включительно:
|   j = 0
|   если a[i;j] > 0 то: pol = a[i;j]
|                       ans_i = i
|                       ans_j = j
|   os = 1
|   НЦ пока i+os < N и j+os < N:
|   |   если a[i+os;j+os] > 0 то: pol = a[i;j]
|   |                             ans_i = i+os
|   |                             ans_j = j+os
|   |   os = os+1
|   КЦ
КЦ
вывод ans_i, ans_j
Конец.
```

(Проход по части матрицы выше главной диагонали и по главной диагонали)

(Проход по части матрицы, которая ниже главной диагонали)

(+)

Олимпиада школьников «Надежда энергетики»

МЭИ, г. Москва

Место проведения

М9 67-14

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ Кугерин

ИМЯ Георгий

ОТЧЕСТВО Дмитриевич

Дата рождения 10.05.2002 **Класс:** 11

Предмет информатика **Этап:** заключительный

Работа выполнена на 9 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: Кугерин

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ Н9 67-14

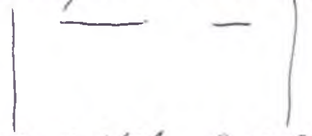
ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

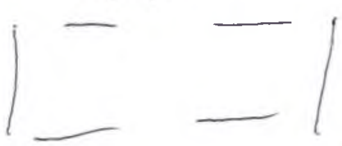
№1.

Черты в верхнем ~~нес~~ ряду принимают только два „значение". Либо они оба пересечены, либо оба непересечены. Значит, чтобы определить состояние черт в верхнем ряду, необходимо на одну из черт верхнего ряда поставить датчик. Если в момент определения буквы верхний датчик ~~оказался~~ не показывает, что черта непересечена, значит это знак . Иначе для однозначного определения символа необходимо показаться датчиками внизу.

Оценка:

Покажем, что 1 датчика внизу поставить недостаточно. Если мы поставим датчик в левый ~~нижний~~ угол, ~~то может быть~~ две ситуации.

~~Если~~ Если датчик ~~не горит~~ показывает, что символ непересечен, это может быть символ  или , только с двумя датчиками определить что однозначно не сможем.

Если мы поставим датчик в правый нижний угол, то у нас опять возникает ситуация, когда мы не сможем точно определить символ: это  и . Поэтому двух датчиков для решения задачи недостаточно

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ 67-14

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Пример:

Покажем, что трёх датчиков для решения задачи достаточно. ~~Но~~ Как уже было сказано ранее, один датчик мы ставим на одну из крайних черт, ~~два~~ для определения позиции знаков сверху, два других снизу. Покажем, что с 3-мя датчиками можно однозначно определить букву. ~~разобрав все случаи.~~ (знаком

~~? далее обозначается то, что положение черт мы не знаем)~~

~~| . ? . | → такой ситуации нет,~~

Если вверху датчик показывает, что черта не пересечена, это точно знак | _ _ |. ~~Если~~ ~~Далее~~ иначе, Если внизу оба датчика непересечены, то это символ | ‾ ‾ |. Если пересечён только один из датчиков — это символ | ‾ ‾ _ |. Если пересечены оба, это символ | = = |. Таким образом, знаки полностью определены.

Ответ: 3 датчика.

№5.

Описание алгоритма

Разобьем алгоритм на две части. Первая из них — это обход двора до точки (N, N) — (считаем нумерацию клеток с 1). При этом обходе есть два вида ходов: ~~у нас~~. ~~Наша~~ ~~абсцисс~~ горизонтальная координата уменьшилась на 1, а вертикальная координата равна 1. Второй тип ~~пр~~ ходов — горизонтальная координата равна N, а вертикальная увеличивается на 1.

После того, как мы пройдём точку (~~5~~N, ~~5~~N), то ~~и~~ у нас также есть два типа ходов. Первый — гориз. координата равна 1, а вертикальная увеличивается на 1 (до N). Второй тип — горизонтальная координата уменьшается на 1, а вертикальная равна N.

На каждом шаге мы считываем число и проверяем положительное ли оно. Если да — сохраняем координаты.

алг Ларец

нач

цел ~~текущая~~ текущГор, ~~те~~ текущВерт, отвГор, отвВер, n, k

~~отв~~ ~~отв текущ~~Гор =

~~пока~~

ввод n

текущГор = n

текущВер = 1

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-14

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

пока ~~X~~ текущГор <> 1 (~~<> - знак не равно~~ ~~идём до точки (1; 1)~~)

нц

ввод k (случай 1)

если k > 0 то

~~Теки~~ отвГор = текущГор

отвВер = ~~текущВер~~ 1

всё

текущГор = текущГор - 1

текущВер = текущВер + 1

ввод k

если k > 0 то (к случай 2)

отвГор = n

отвВер = текущВер

всё

кц

(~~далее обработаем точки (1; 1) и (N; N)~~

далее обработаем точки (1; 1) и (N; N)

и (1; 2) - переходы

ввод k

~~если k > 0~~

если k > 0 то

отвГор = 1

отвВер = 1

всё

~~есл~~ ввод k

если k > 0 то

отвГор = N

отвВер = N

всё

(~~обрабатываем переход от (N; N)~~ 1 2

до (~~N; 1; N~~))

~~[illegible]~~ ввод k

если k > 0 то

отвГор = 1

отвВер = 2

Лист 04 из 09

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-14

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

~~нач~~ ~~текущГор = 1~~ кё

~~текущВер = 2~~ текущГор = N − 1

текущВер = ~~N−1~~ 2

~~пока (текущ Гор~~

пока (текущГор <> 2) ~~т~~

нц

Ввод k

если ~~k~~ k > 0 то

отвГор = текущГор

отвВер = N

кё

текущГор = текущГор − 1

~~т~~екущВер = ~~т~~ текущВер + 1

Ввод k

если k > 0 то

отвГор = 1

отвВер = текущВер

всё

кц.

(Нам осталось обработать два последних значени: (2; N) и (1; N))

Ввод k

если k > 0 то

отвГор = 2

отвВер = N

кё

Ввод k

если k > 0 то

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-14

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

отвТор = 1
отвВер = N
кц
Вывод отвТор, отвВер
конец

№2.

Т.к. требуется сложность O(N²), достаточно отсортировать нам массив, а затем найти в нём k-е по возрастанию, перебирая эл-ты отсортированного массива.

алг Возрастание
нач
цел N, отв, i, j, t a[100000], счетчик, k
~~Ввод~~ N
~~Ввод~~ k
для ~~N~~ i от 1 до N
нц
Ввод a[i]
кц
для i от 1 до N-1 (сортировка пузырьком)
нц для j от 1 до N-i+1
нц
если a[i] > a[j] то
t = a[i]
a[i] = a[j]
a[j] = t
кв
кц
кц

+

(Внимание! Далее подразумевается, что k-ое по возрастанию число является различным, например в списке 1, 1, 2, 2 2-е по возр. число 2)

Лист 6 из 07

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЭ 67-14

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

об=a[1]

для i от 2 до N

нц

если ~~(~~a[i-1] <> a[i]) то

счётчик = счётчик + 1

если счётчик = k то

об = a[k]

выход (выход-аналог break)

кц

кц

вывод об

конец

№3.

Будем хранить данные о брёвнах в одномерном массиве. ~~Если~~ Для каждой абсциссы сохраним количество брёвен с этой абсциссой. Для нахождения ответа найдём максимум в этом массиве. Чтобы не хранить большие числа, приведём все к началу координат (1;1)

алг брёвна

начало

цел a[100 000], ~~нач коорд~~, xнач, yнач,

~~x~~ n, i, x, y, макс Абс, об

Ввод n

Ввод xнач, yнач

максАбс = 1

для i от 1 до N нц

Ввод x, y

доп лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ 67-14

$a[1+(x-x_{нач})] = a[1+(x-x_{нач})]+1$

~~если x > x макс то~~ если $(x - x_{нач}) >$ макс Абс (определениеупр. массива)

макс Абс $= x - x_{нач} + 1$

кц

отв $= a[1]$

для i от 2 до макс Абс

нц

если $a[i] >$ отв то

отв $= a[i]$

все

кц

вывод отв

конец

№4

~~Заметим, что любое натуральное число можно представить в виде суммы неповторяющихся чисел в троичной системе счисления, состоящих из цифр 0 или 1. Поэтому для того, что взвесить груз весом от 1 до 120 кг, нам необходимы гири с весами 3^1, 3^2, 3^3,~~ ~~3^4, 3^5 (гири четырех гирь будет недостаточно, или же что отмерить максимум 3^3 (3+~~

~~3^1, 3^2, 3^3, 3^4. Трёх гирь будет недостаточно, так как ими можно отмерить максимум $27+9+3=39$ кг.) Четырьмя же гирями можно максимум отмерить $81+27+9+3=120$ кг, как раз хватает. Пятая гиря (вес макс. гири 243 — избыточно). Поэтому:~~

~~Ответ: гири: 3; 9; 27~~

Лист 08 из 09

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

~~1, 3~~, 3^0, 3^1, 3^2, 3^3, 3^4). ~~Четырь~~ Комплекта гирь (3^0, 3^1, 3^2, 3^3) недостаточно, так как ими можно отмерить максимум $27+9+3+1=40$ кг. В это же время, гиря с массой $3^5=243$ кг избыточна: максимум комплектом из 5 гирь можно отмерить $81+27+9+3+1=121$ кг. Поэтому <u>Ответ</u>: 5 гирь, ~~не~~ веса 1, 3, 9, 27, 81 кг.

Олимпиада школьников «Надежда энергетики»

МЭИ г. Москва

Место проведения

НЯ 67-84

← **Не заполнять**
Заполняется ответственным работником

шифр

Вариант № 73111

ФАМИЛИЯ ЛЕВИЦКИЙ

ИМЯ ДАНИИЛ

ОТЧЕСТВО РОМАНОВИЧ

Дата рождения 07.07.2002 **Класс:** 11

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 5 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

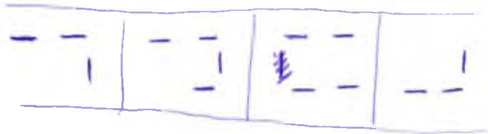
Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача N1

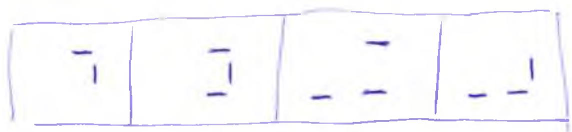
Чтобы узнать, сколько датчиков (минимально) необходимо поставить, будем пошагово отсеивать черточки ~~в буквах~~, исчезновение которых не будет нарушать разницу в символах.

Во-первых, заметим, что во всех символах левая черта нарисована. Значит, мы точно можем её не считывать. Перерисуем алфавит не обращая на неё внимание:

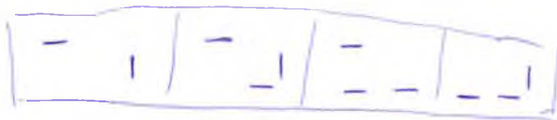


Если попытаться „убрать" одну из верхних черточек, символы все ещё будут различимы:

ПРИМЕР 1



ПРИМЕР 2



~~Вы~~ Значит, ~~один~~ одну из этих черточек можно не подвергать идентификации.

Так же мы можем ~~в~~ убрать и ~~левую~~ нижнюю черту (алфавит не пострадает)

Но дальнейшие попытки убрать черточки приведут к неразличимости символов. Значит, бояре должны добыть 3 датчика (для верхней-правой либо верхней левой, для правой и для правой-нижней черточек)

±

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача №2

Представив колонку в виде массива размером N, отсортирую её „пузырьковым" методом, сложность которого как раз сравнима с N^2. Затем выведу элемент массива с индексом K.

Переменные:

```
  массив a[N]
  целые K, temp
  для i от 1 до N:
      Ввод(a[i])
  Ввод(K)
  Если K ≤ 0 или K > N:
      Вывести(некорректный ввод)
  Иначе:
      для i от 1 до N:
          для j от 1 до N-i:
              если a[j] > a[j+1] то:
                  temp = a[j+1]
                  a[j+1] = a[j]
                  a[j] = temp
              ~~конец „если"~~
          ~~конец „для"~~
      ~~конец „для"~~
      Вывести(a[K])
```

±

Задача №3

Представим ущелье как матрицу N*n, где N - длина ущелья. Бревна в данной матрице будут представлены в виде единиц (если бревна нет, значение элемента равно нулю). Причем единицы начинают встречаться в матрице только с k-той строки (участка). ? Необходимо узнать индекс строки, в которой содержится больше всего единиц. Для этого заведем переменные max, куда будем записывать кол-во единиц и max_index, где будет храниться номер участка с макс. кол-вом бревен

Переменные:
~~массив~~ целые N, n, max, max_index, K, counter
матрица a[N;n]

```
~~для~~
Ввод(K)
max_index = 0
max = 0
для i от K до N :
    counter = 0
    для j от 1 до n :
        ~~ес~~
        ~~сон~~
        если a[i;j] = 1 :
            counter = counter + 1
    если counter > max :
        max = counter
        max_index = i
Вывести (max_index)
```

Задача 4

Рассмотрим сперва $Q=40$ т.к. у нас всего 4 груза, должно выполняться неравенство: $x_1+x_2+x_3+x_4 \geq 40$

Попробуем найти веса этих грузов следующим образом: начиная с 1кг будем моделировать равновесие на весах. Когда грузиков станет нехватать, будем ~~добавлять~~ представлять следующую переменную как макс. возможный вес.

$Q=1$ $\quad x_1=1 \quad x_2=3 \quad x_3=9$

① | $\underline{\underline{1}}$

②+1 | $\underline{\underline{3}}$

③ | 3

④ | 3+1

⑤+1+3 | $\underline{\underline{9}}$

=> Отсюда заметим, что веса ~~и грузиков~~ гирь изменяются как степени тройки. То есть $x_4=27$. Действительно, данные веса позволяют взвесить любой груз в диапазоне от 1 до 40, причем условие $x_1+x_2+x_3+x_4 \geq 40$ выполняется

Следуя данной логике, для $Q=120$ кг понадобится ~~груз~~ гиря x_5, который будет равен 81 (Меньше грузов взять нельзя т.к. тогда не будет соблюдаться условие, что веса всех гирь ≥ 120

Ответ: число гирь = 5 ; веса гирь: 1, 3, 9, 27, 81

(+)

Задача 5

Создадим матрицу и по мере ее обхода будем ~~записывать элементы в массив размера NxN. Затем пройдем по нему от конца до начала, пока не встретится положительный элемент~~ определять положит. элементы и записывать их ~~в отде~~ координаты в отдельную переменную.

Доп. лист

Задача 5 ~~(продолжение)~~ (продолжение)

Переменные:

```
матрица matrix[N;N]
~~массив a[N*N]~~
целые i,x,y, ~~count~~ plusx, plusy

plusx=0
plusy=0
для i от 1 до N: # первая половина таблицы
|   для x от N-i+1 до N:
|   |   для y от 1 до i:
|   |   |   если matrix[x;y]>0:
|   |   |   |   ~~plus=matrix[x;y]~~
|   |   |   |   plusx=x
|   |   |   |   plusy=y
для i от 1 до N: # вторая половина
|   для x от 1 до N-i:
|   |   для y от 1+i до N:
|   |   |   если matrix[x;y]>0:
|   |   |   |   ~~plus=matrix[x;y]~~
|   |   |   |   plusx=x
|   |   |   |   plusy=y
~~если plus=0:~~
~~    вывод(нет положительных)~~
~~иначе~~
если plusx и plusy=0:
|   вывод(нет положительных)
иначе:
|   вывод(plusx)
|   вывод(plusy)
```

+

Олимпиада школьников «Надежда энергетики»

сош №20,
г. Новочебоксарск

Место проведения

PN 69-59

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ Лукин

ИМЯ Максим

ОТЧЕСТВО Сергеевич

Дата рождения 17.05.2003 **Класс:** 10

Предмет информатика **Этап:** заключительный

Работа выполнена на 3 **листах** **Дата выполнения работы:** 16.02.20
(число, месяц, год)

Подпись участника олимпиады: Лукин

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ PN 69-59

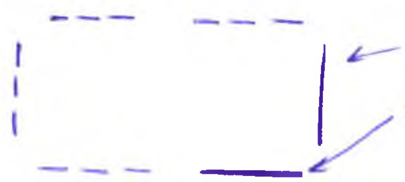
ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1.

Если бояре добудут 1 датчик, то невозможно будет отличить 4 знака, возможно будет отличить только 2 (1: либо датчик горит; 2: либо датчик не горит).

Если бояре добудут 2 датчика, то возможно будет определить 4 знака:
1) ни один не горит;
2) горит только первый;
3) горит только второй;
4) горят оба.

Приведу пример (—— - есть датчик; - - - - нет датчика)



1: Если не горит ни один датчик, это знак .

2: Если горит только первый датчик, это знак .

3: Если горит только второй датчик, это знак 

4: Если горят оба датчика, это знак 

Ответ: для работы необходимо 2 датчика.

(+)

№2

Чтобы веса удовлетворяли условиям: следующее число должно быть на 1 больше удвоенной суммы всех написанных до того момента чисел. ~~Все~~ Тогда мы не сможем получить один вес 2 раза.

Этому условию удовлетворяют три весом 1, 3, 9, 27.

$3 = 1 \cdot 2 + 1$, $9 = (1+3)2+1$, $27 = (1+3+9)2+1$.

(±)

Докажу, что с их помощью можно получить любой вес (в целом числе кг) от 1 до 40.

1: 1	6: 9−3	11: 9+3−1	16: 27−9−3+1	21: 27−9+3	26: 27−1	31: 27+3+1	36: 27+9
2: 3−1	7: 9−3+1	12: 9+3	17: 27−9−1	22: 27−9+3+1	27: 27	32: 27+9−3−1	37: 27+9+1
3: 3	8: 9−1	13: 9+3+1	18: 27−9	23: 27−3−1	28: 27+1	33: 27+9−3	38: 27+9+3−1
4: 3+1	9: 9	14: 27−9−3−1	19: 27−9+1	24: 27−3	29: 27+3−1	34: 27+9−3+1	39: 27+9+3
5: 9−3−1	10: 9+1	15: 27−9−3	20: 27−9+3−1	25: 27−3+1	30: 27+3	35: 27+9−1	40: 27+9+3+1

С помощью 4 гирь нельзя получить число > 40, т.к. $27+9+3+1 = 40$ => нужна 5 гиря. Аналогично: вес 5-ой гири $= (1+3+9+27)2+1 = 81$.

С помощью вычитаний из 81, можем получить любое целое число от 41 до 80, т.к. мы можем получить любое целое число от 1 до 40. Путем сложения ~~его~~ с гирей весом 81 → от 1 до 9 кг получим веса от 82 до 90. (Чтобы получить 81 достаточно 1 гири весом 81).

Итак, с помощью 5 гирь весом 1 кг, 3 кг, 9 кг, 27 кг и 81 кг можем получить все веса от 1 до 121 кг (1+3+9+27+81=121 кг)
121 кг > W = 90кг => условие выполняется.

Ответ: минимальное число гирь = 5. (1кг; 3кг; 9кг; 27кг и 81кг)

№3

```
1  for(int i; i<=n; i++)
2  {
3    if (i%2 == 1)
4    {
5      for(int j=1; j<=n; j++)
6      {
7        if (a[i][j]>=d && a[i][j]<=s)
8        {
9          cout << i << " " << j;
10         return 0;
11       }
12     }
13   }
14   if (i%2==0)
15   {
16     for (int j=n; j>=1; j--)
17     {
18       if (a[i][g] >=d && a[i][j]<=s)
19       {
20         cout << i << " " << j;
21         return 0;
22       }
23     }
24   }
25 }
```

Алг + прогр

n>0

Ввод?

d≤s

+

Пояснение 1-25: цикл, проходящий по всей матрице.
3: проверяем номер строки на нечетность;
4-13: цикл, работающий, если номер строки нечетный
5: цикл, проходящий по строке слева направо (от 1 до N элемента)
7: проверяем число в матрице на принадлежность к отрезку [d; S]
9: если число принадлежит отрезку, выводим номера строки и столбца
10: ——— || ———, заканчиваем программу.
14: проверяем номер строки на четность
15-24: цикл, работающий, если номер строки четный;
16: цикл, проходящий по строке справа налево (от N до 1 элемента)
18: проверяем число в матрице на принадлежность к отрезку [d; S]
20: если число принадлежит отрезку, выводим номер строки и столбца
21: ——— || ———, заканчиваем программу.

Выше приведен только алгоритм поиска нужного числа, перед которым [алгоритмом], разумеется, необходимо объявить переменные.

Ответ: алгоритм описанный выше.

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ PN 69-59

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№4

1) Общая формула для слагаемого с номером N.

$S_N = -(-1)^n \cdot \frac{x^{2n-1}}{N}$.

$x^{2n-1} \to x$ в степени, которая с каждым новым N увеличивается на 2.

$N \to$ число в знаменателе, равное номеру элемента.

$-(-1)^n$, если n нечетное, то $(-1)^n \to$ ~~...~~ отрицательно $= -1$, тогда $-(-1) = 1$, что и требуется: $(+S_N)$.

— если n четное, то $(-1)^n \to$ положительное $=1$, тогда $-(1) = -1$, что и требуется $(-S_N)$.

2) Формула из условия $S_{N+1} = -x^2 \cdot \frac{N}{N+1} \cdot S_N$

Кол-во арифметических операций:

1) $N+1$; 2) $\frac{N}{N+1}$; 3) x^2; 4) $-x^2$, 5) $-x^2 \cdot \frac{N}{N+1}$; 6) $-x^2 \frac{N}{N+1} \cdot S_N$.

Кол-во арифметических операций в моей формуле:

1) $2n$; 2) $2n-1$; 3) x^{2n-1}; 4) $\frac{x^{2n-1}}{N}$; 5) $(-1)^n$; 6) $-(-1)^n$; 7) $-(-1)^n \frac{x^{2n-1}}{N}$

В моей формуле на одну арифметическую операцию больше, однако ее преимущество в том, что достаточно знать только номер элемента (N), а в формуле из условия необходимо также знать значение S_{N-1}.

Ответ: 1) $-(-1)^n \cdot \frac{x^{2n-1}}{N}$; 2) кол-во операций в моей формуле (7), больше кол-ва операций в формуле из условия (6).

№5.

Множество $Z_m = [0, 1, 2, \ldots, m-1] \to$ все возможные остатки от деления на m

1) Найдем D. $D = b*b - 4*A*C$.

2) Проверим D ~~на~~ (Если $D \geq 0$ продолжаем, если $D < 0 \to$ код ненадежный)

3) Если $D \geq 0$ найдем корни уравнения: $x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$

Итак: код является надежным, если D — является ~~корн~~ квадратом целого числа, если $-b+\sqrt{D}$ или $-b-\sqrt{D}$ делится на $2a$ без остатка (хотя бы одно). Если оба условия соблюдены, проверяем корни, которые подошли под оба условия. Если данный корень не меньше 0 и меньше m, то числа для замка являются надежными кодами.

Ответ: алгоритм, описанный выше.

Лист 03 из 03

Олимпиада школьников «Надежда энергетики»

МЭИ город Москва

Место проведения

НЯ 64-58

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ СЕРГЕЕВ

ИМЯ ИЛЬЯ

ОТЧЕСТВО ИВАНОВИЧ

Дата рождения 22.08.2002

Класс: 11

Предмет ИНФОРМАТИКА

Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 5 листах

Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

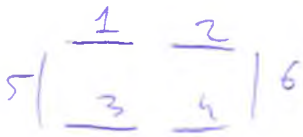
ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1: Виды знаков:

Заметим, что у каждого знака есть левая вертикальная черта (|). Из этого следует то, что датчик для неё не нужен.

Предположим, что минимальное требуемое кол-во датчиков равно двум. Тогда ~~свой~~ первый датчик должен проверять наличие черты, которая имеется лишь у половины символов, а второй должен проверять наличие черты, которая имеется у половины символов, отобранных первым датчиком. Т. е проверив наличие одной черты, мы должны уменьшить количество символов, которыми может быть рассматриваемый символ до двух, а затем по одной оставшейся черте понять, какой же знак мы пытаемся опознать.

Обозначим черты:



Черта 3 есть лишь у половины знаков, больше таких черт нету. У тех знаков, у которых нету черты 3 общей (той, которая есть у обоих) не явл. черта 4 (и только она). А у тех, у которых черта 3 есть общими не явл. черты 1 и 2. => мы не можем поставить второй датчик так, чтобы он помогал определить знак в обоих случаях.

Также очевидно, что одним датчиком невозможно обойтись, т. к для каждой черты есть всего два варианта: она присутствует, или нет. ~~на~~ А знаков у нас 4.

=> Минимальное кол-во датчиков = 3 (прим: датчики для черт 3, 4 и 2)

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

к заданию 1: Ответ: 3 +

№ 2: Опишу алгоритм на естественном языке:

1. Считаем (узнаем N и K)
2. Отсортируем числа по возрастанию
3. Выведем K-ое число.

~~Так~~ В таком случае, если мы используем в пункте 2 сортировку слиянием, то суммарное кол-во операций будет $\approx N \cdot \log_2 N$;
Но нам нужно кол-во операций "порядка N^2", поэтому в пункте 2 используем сортировку выбором.
Пояснение: сортировка выбором совершает N шагов на каждом шаге выбирается минимальное число из последних ~~N-i+1~~ $N-i+1$ чисел, где i - номер шага, считая с единицы.
В конце каждого шага минимальное число переставляют местами с числом под номером i.
Т.к ~~для~~ на каждом шаге алгоритм просматривает $N-i+1$ чисел, то асимптотически кол-во совершаемых операций $\approx N^2$. +

№ 5: Опишу алгоритм на естественном языке:

1. Введём (узнаем) N вещ
2. Будем двигаться по диагонали и запоминать номера столбца и строки, в которых был последний положительный эл-т.
Если встречаем при дв-ии положительный эл-т, то перезаписываем эта номера.
~~[illegible]~~

к заданию 5:

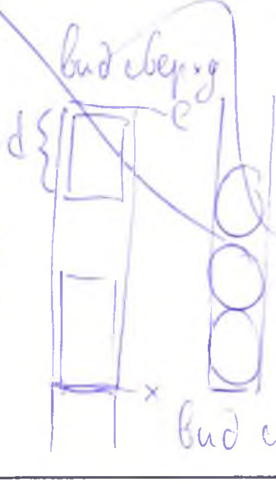
3. Выведем номера столбца и строки, в которых был последний положительный эл-т.

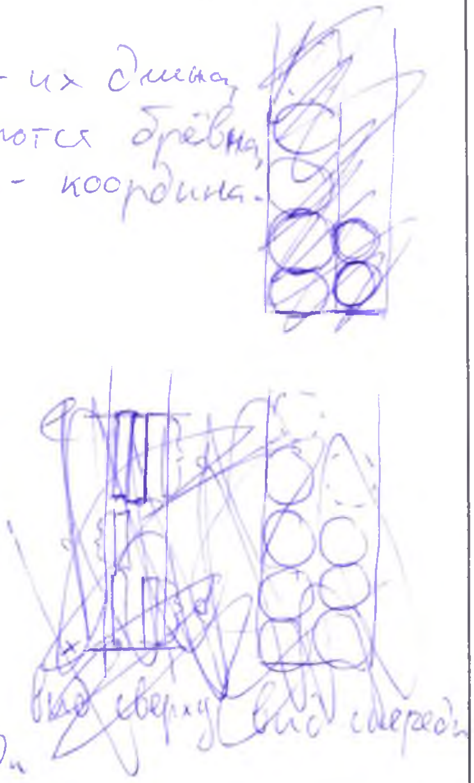
Пояснение к пункту 2:

Пусть стартовая клетка имеет координаты $(N;N)$. Тогда из неё мы перейдём в клетку с координатами $(N-1;N)$ и начнём движение по "диагонали", т.е. будем ~~уменьшать~~ ~~уходить~~ переходить в клетку с i, меньшим на 1, и j, большим на единицу, если такая клетка находится в границах поля. В противном случае посмотрим на ~~перейдём в~~ клетку $(i-1;j)$, где $(i;j)$ - клетка, с которой мы начали движение по диагонали. ~~[illegible]~~ Если $i-1>0$, то перейдём в неё и повторим весь процесс. В противном случае перейдём в клетку $(i;j-1)$.

(+)

~~№3: Опишу алгоритм на естественном языке:~~
~~Пусть a – текущая координата, на которую смотрит Горыныч.~~
~~Посмотрим на брёвна: пусть d – их длина, x – координата, в которой начинаются брёвна, а l – длина ущелья (и она же – координата конца ущелья).~~
~~Примем за k количество брёвен в виде сверху (т.е. кол-во стопок лежащих друг на друге брёвен)~~





Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-58

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

~~К задаче 3: Если длина бревна = d, то k = l/d. Тогда рассмотрим все k "стопочек" из брёвен и выберем "стопочку" с максимальной шириной. Это и будет максимальная ширина завала.~~

№4:

№3: Опишу алгоритм на естественном языке:

1. Введём координаты начал брёвен и длину угла. Создадим переменную для ответа ans = 1

2. Отсортируем все координаты начал.

3. Введём (создадим) два указателя: i=1 и j=1. Будем передвигать i на +1, пока i < n и координата начала с номером i не равна координате начала с номером i+1. В противном случае приравниваем i = j и начинаем двигать j на +1, пока j < n и координата начала с номером j = координате начала с номером ~~j+1. В противном случае приравниваем i=j и начинаем шаг с начала, если i < n иначе~~

4. ... с номером j+1. В противном случае ~~приравниваем~~ приравняем переменную для ответа к максимуму из неё и j − i + 1. Далее приравняем i = j, и если i < n, то выполним шаг 4 ещё раз.

5. Выведем ans.

Пояснение к пункту 2: для сортировки используем известную сортировку слиянием. (Сортировка, в которой мы рекурсивно сортируем половины массива, а затем сливаем их в одно целое методом двух указателей)

(7)

Доп. лист.

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НЯ 67-58

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

~~№4: Используем то, что мы знаем, как можно взвесить груз от 1 до 40 килограмм с 4-мя гирями.~~
~~Вес от 1 до 120 кг мы бы могли взвесить $\frac{120}{40}$~~
~~12 гирь. Попытаемся улучшить ответ.~~
~~Мы можем дополнительно использовать еще~~

~~Примечание по условию: в условии сказано, что требуется взвесить груз в 120 кг. Т.е если воспринимать задачу напрямую, достаточно будет одной гири в 120 кг для этого. Т.е ответ в таком случае: 1 гиря - 120 кг~~

Если же имеется в виду то, что требуется уметь взвешивать любой груз от 1 до 120 кг., то достаточно будет добавить две гири по 40 кг. Тогда, если вес > 40 кг, то мы добавим на чашу весов без груза гирю в 40 кг, и проделаем всё те же операции, как если бы мы взвешивали Q - 40 кг. Если же и тогда не хватит гирь и чаша весов с Q будет перевешивать, добавим на чашу без груза еще одну гирю в 40 кг. и проделаем всё то же, что мы делали, когда взвешивали груз от 1 до 40 кг. (т.к текущий вес груза будет > 80 и < 120, то Q - 80 ∈ [1; 40] и мы сможем так сделать).
Можно заметить, что с гирями весами в 2, 5, 10 и 20 кг можно будет взвешивать любой груз от 1 до 20кг.

+

Ответ: 6 гирь: 2, 5, 10, 20, 40, 40 кг.

Олимпиада школьников «Надежда энергетики»

МЭИ город Москва

Место проведения

BS 83-24

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ СКЛЯРЕНКО

ИМЯ ЕВГЕНИЙ

ОТЧЕСТВО АНАТОЛЬЕВИЧ

Дата рождения 08.11.2003 **Класс:** 10

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 **листах** **Дата выполнения работы:** 16.02.20
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 88 - 24

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

N3

Если sp - исходная матрица размерами N·N,

то:

i=0
flag=False
пока i<n:

om j=0 go j=n-1 {
если (sp[i][j] ≤5 и sp[i][j] ≥ d), то
flag =True
ind1=i
ind2=j
„выход из внутреннего цикла" (break)

если(flag == True) то
„выход из внешнего цикла"(break)

иначе:
если i==n, то „выход из внешнего цикла" (break)
~~om~~ i+=1
om j=n-1 go j=0
если (sp[i][j] ≤5 и sp[i][j] ≥d, то
flag = True
ind1=i
ind2= j-n+1 - „т.к. мы шли с конца"
„выход из внутреннего цикла" (break)

если (flag== True) то
„выход из внешнего цикла" (break)
иначе: {
i+=1

n=N?

N>0

d≤5?

±

Выводим ind1 и ind2.

Объяснение алгоритма:
flag - переменная, используемая для проверки, нашли ли мы уже нужный элемент".
„чётные строки матрицы { в 0 индексации!"
Цикл от j=0 до j=n-1 идёт слева направо, поэтому ответ для элемента, найденного в таком цикле - (i j)

Цикл от j=n-1 до j=0 идёт справа налево (нечётные строки матрицы в 0 индексации), поэтому ответ для элемента, найденного в таком цикле равен (i j-n+1)
Программа подразумевает, что такой элемент в матрице точно есть, иначе будет ошибка

$S(x) = x - \frac{x^3}{2} + \frac{x^5}{3} - \frac{x^7}{4} + \dots$

1) $S(x)_N = (-1)^{N\%2-1} \cdot \frac{x^{2N-1}}{N}$; тогда

второй элемент равен: $(-1)^1 \cdot \frac{x^3}{2} = -\frac{x^3}{2}$

третий: $(-1)^2 \cdot \frac{x^5}{3} = \frac{x^5}{3}$

четвёртый: $(-1)^3 \cdot \frac{x^7}{4} = -\frac{x^7}{4}$

первый: $(-1)^0 \cdot \frac{x}{1} = x$ Ответ: $(-1)^{N\%2-1} \cdot \frac{x^{2N-1}}{N}$

2) ~~Фор~~ ~~В~~ Формула $(-1)^{N\%2-1} \cdot \frac{x^{2N-1}}{N}$ имеет общую асимптотику $O(2N-1)$, т.к. x^{2N-1} находится за $2N-1$ операцию. Остальные операции делаются за $O(1)$.

Формула $\frac{S_{N+1}}{S_N} = -x^2 \cdot \frac{N}{N+1}$ имеет общую асимптотику $O(N+1)$, т.к. сумма S_{N+1} находится за $(N+1)$, а если сохранять ~~в~~ где-то все предыдущие посчитанные суммы, то сумма S_N уже будет посчитана, когда мы найдём S_{N+1}. (~~такой же~~ Динамическое программирование).

+

Значит в формуле из условия (пункта 2) количество затрачиваемых операций меньше.

Дополнительные ~~о~~ пояснения к алгоритму задания 3:

1) Цикл ~~пока~~ (пока i<n:) идёт до последней строки матрицы, а затем прекращается.

2) Проверка внутри цикла „пока" (если i==n) нужна в случае, если количество строк матрицы нечётно и мы перейдём к n-ой строке матрицы после выполнения первого цикла „от" внутри цикла „пока"

3) Операция „break" нужна для прекращения работы цикла, выхода из него.

4) Если в матрице не будет элемента, удовлетворяющего условию, то элементы ind1 и ind2 не будут существовать и программа выдаст ошибку. (Подразумевается, что такой элемент существует в матрице.

№5.

Уравнение $ax^2+bx+c=0$, если $a\neq 0$ имеет два решения:

$$\begin{cases} x_1 = \dfrac{-b+\sqrt{b^2-4ac}}{2a} \\ x_2 = \dfrac{-b-\sqrt{b^2-4ac}}{2a} \end{cases}$$

Проверим лежит ли хотя бы один из этих корней в отрезке от 1 до $m-1$: $[1; m-1]$. Если да, то код надёжный, если нет, то ненадёжный.

Если же $a=0$, то уравнение принимает вид $bx+c=0$

Решение одно $x=-\frac{c}{b}$.

Если $-\frac{c}{b}$ входит в отрезок $[1; m-1]$, то код опять же надёжный, если нет, то ненадёжный.

+

№2.

Взяв, например, гири $2|5|15|45$ мы можем взвесить любой груз до 67, т.к. $45+15+5+2=67$

1) Возьмём гирю 2кг, мы можем, используя только её, взвесить все грузы $[1;2]$, пользуясь $(> < =)$

2) Гири 2 и 5, все гири до 7 $[1;7]$, также пользуясь только $(> < =)$

И так далее. Несложно заметить, что таким образом мы можем взвесить все грузы до суммы всех имеющихся гирь.

Так как четырёх гирь достаточно только для взвешивания грузов до 67 кг, а наибольшая из этих гирь $=45$, то следующая гиря будет точно больше, чем $67+45 > 90$, значит, используя 5 гирь, мы сможем взвесить любой груз от $[1; 90]$. Набор гирь: $2|5|15|45|2+5+15+45+67=135|$

Ответ: 5 гирь; 2кг|5кг|15кг|45кг|135кг|.

−

Доп. лист

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 88-24

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1

Видно, что нет смысла проверять три черты: две верхние и одну боковую левую, т.к. они присутствуют во всех символах.

Один датчик нам нужен, чтобы проверить есть ли черта на правой боковой стороне.

Вне зависимости от того, есть ли черта на правой боковой стороне, нам нужен ещё один датчик на правую нижнюю сторону. Тогда алгоритм работы этих двух датчиков по различию четырёх символов таков:

1) Если есть черта на правой боковой стороне и есть черта на правой нижней стороне, то это символ 2-ого типа.

2) Если есть черта на правой боковой стороне и нет черты на нижней правой стороне, то это символ 1-ого типа

3) Если черты нет ни там ни там, то это символ 4-ого типа.

4) Если черты нет на правой боковой стороне, но она есть на нижней правой стороне, то это символ 3-го типа.

+

Олимпиада школьников «Надежда энергетики»

СФ МЭИ

Место проведения

NC 75-12

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ Соколовская

ИМЯ Арина

ОТЧЕСТВО Владимировна

Дата рождения 13.09.2003 Класс: 10

Предмет Информатика Этап: Заключительный

Работа выполнена на 3 листах Дата выполнения работы: 18 февраля 2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

1.

Изначально датчики отсутствуют, значит для работы первого знака нам понадобится 4 новых датчика.

2) Для работы второго знака у нас уже есть и нужных датчика, значит нужно купить ещё один.

3) Для работы третьего знака есть и нужных датчика, то надо купить ещё 1.

4) Для работы четвертого знака есть все нужные датчики.

То есть для работы системы нужно 4+1+1 =6 датчиков.

Ответ 6 датчиков.

N>0

3.
```
читаем N;
читаем таблицу;
x = -1, y = -1;
for i = от 1 до N
  Если i нечётное, то
    for j = от 1 до N
      Если (a[i][j] >= d и a[i][j] <= s), то
        Если x != -1, то x = i;
                         y = j;
  Иначе
    for j = N до 1
      Если (a[i][j] >= d и a[i][j] <= s), то
        Если x != -1, то x = i
                         y = j;
Вывести x, y;
end.
```

Олимпиада школьников «Надежда энергетики»

Вариант: 73101 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ NC 75-12

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

5. читаем a, b, c, m; f=0;
for i = от 0 до m-1
Если (a*i*i + b*i+c=0), то [f=1, break;]
Если f=1, то решения есть
иначе решения нет

+

4. 1) $a_n = \frac{x^{2*n-1}}{n}$, если n - чётное, то домножить на -1.

2) 1. $\frac{S_{N+1}}{S_N} = -x^2 * \frac{N}{N+1}$; 2. $a_N = \frac{x^{2N-1}}{N}$

+

В первой формуле понадобится несколько арифметических операций, которые выполняются за O(1). Во второй возведение в степень потребует двоичный логарифм от 2*N-1 (бинарное возведение в степень)

2. 5 гирь весом 1 3 9 27 81

$\frac{1}{1}$ $\frac{2}{3-1}$ $\frac{3}{3}$ $\frac{4}{3+1}$ $\frac{5}{9-3-1}$ $\frac{6}{9-3}$ $\frac{7}{9-3+1}$ $\frac{8}{9-1}$ $\frac{9}{9}$ $\frac{10}{9+1}$ $\frac{11}{9+3-1}$
$\frac{12}{9+3}$ $\frac{13}{9+3+1}$

+

Из первых трех гирь мы можем собрать все веса от 1 до 13, то используя еще и гирю весом 27 мы сможем набрать также веса от 27-13(14) до 27+13(40). Теперь мы можем набрать все веса от 1 до. Используя еще гирю весом 81 мы можем набрать веса от 81-40(41) до 81+40(121).

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Олимпиада школьников «Надежда энергетики»

г. Гусь-Хрустальный
МБОУ „СОШ №2"

Место проведения

RX 76-61

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ Сухов В

ИМЯ Владимир

ОТЧЕСТВО Игоревич

Дата рождения 21.05.2002

Класс: 11

Предмет ИНФОРМАТИКА

Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 8 **листах**

Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: Сух

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Задание №4

Для решения данной задачи сначала посмотрим, каким образом оптимальнее всего набирать числа.

1) Пусть у нас есть набор чисел: 1, 2, 3

Или мы можем измерить числа: 1, 2, 3, 4, 5, 6.

Следующее число которое нам хотелось бы получить – это 7.

2) Если мы его добавим, то сможем измерить числа:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13. Но давайте добавим не 7, а число 13! Так как мы можем ставить гири на обе части весов, то мы сможем взвесить 7 таким образом

$\underbrace{1, 2, 3, x}_{1\ чаша}$ $\underbrace{13}_{2\ чаша}$, где x – взвешиваемое число 7.

$x = 13 - (1+2+3)$ $7 = 13 - 6$. $13 = 2 \cdot 6 + 1$.

3) Итак, добавив к набору 1, 2, 3 число 13 мы сможем взвешивать числа: 1, 2, 3, ..., 13, 14, 15, 16, 17, 18, 19.

Согласитесь, если у нас в наборе есть 1, 2, 3, 13 и мы можем взвешивать 1...6, то мы можем взвесить и 13-6; 13-5; 13-4; 13-3; 13-2; 13-1.

4) Получаем формулу:

Для набора $a_1, a_2; a_3; ... ; a_n$. $a_{n+1} = (a_1 + a_2 + ... + a_n) \cdot 2 + 1$ $(n \in N)$

5) Тогда для $Q = 40$ нужно 4 гири:

1; 3; 9; 27.

А для $Q = 120$ нужно 5 гирь:

1; 3; 9; 27; 81.

Ответ: 5 гирь: 1, 3, 9, 27, 81

Задание N2
Имеем массив A длины N. Проходимся по нему циклом обновляя самое большое встретившееся число. Как только встретим k-ое число, то выведем его, остановив цикл.
– это первое решение задачи если подразумевается, что мы ищем подпоследовательность с первого элемента.
То есть, для массива A = [1, 3, 4, 1, 2, 3, 4, 5], K = 3
ответ будет 4 (подпоследовательность 1, 3, 4)
В таком случае достаточно будет порядка N операций.

алг задача 2.1
начало алгоритма
цел:
A [const]
цел: N, K, i, t, f

на N ограничений нет.
поэтому const – какое-то постоянное число.
const = N

t = 0
f = 0
ввод N, K
если (N <= 0) или (N < K), то:
вывод "неккоректное N"
всё

Иначе задача не имела бы смысла

если (N > 0) и (N >= K), то:
для i от 1 до N:
нц
ввод A[i]
кц
t = A[1]
f = 1

```
для i от 2 до n:
нц
  если A[i] > t, то:
    f = f + 1
    t = A[i]
  если f = k, то:
    break          ← остановка цикла, когда нашли k-ое число
  все
все
кц
Если k = 1 то:
  вывод A[1]
иначе если f = k
  вывод A[f]
иначе
  вывод "В последовательности не найдено k-ого элемента"
все
все
конец алгоритма
```

2-ое решение. Если подразумевается, что подпоследовательность начинается не обязательно с 1-го элемента массива. Тогда циклом по i будем перебирать откуда начинается подпоследовательность, а циклом по j переберём числа этой подпоследовательности. Получим порядка N^2 операций. Каждый раз запоминаем последнее встретившееся наиб число в последоват.

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RX 76-61

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

алг задача 2.2
начало алгоритма
    цел: A[const]   const = N
    цел: N, k, i, j, t, f, ans
    ans = 0
    t = 0
    f = 0
    ввод N, k
    если (N<=0) или (N<k) то: ← иначе задача не имела бы смысла
        вывод "некорректное N"
    все
    если N>0 ~~...~~ и N>=k то:
        для i от 1 до n:
        нц ввод A[i]
        кц
        ~~t=A[1]~~ ans = A[1]
        f = 1
        для i от 2 до n
        нц t=A[i]; f=1
            для j от i до n
            нц
                если A[j] >t то:
                    f=f+1
                    t=A[j]
                    если f=k то:
                        break   ← остановка цикла по j.
                    все
                все
            кц
            если f=k то:
                ans = A[f]
            все
        кц
        если k=1 то:
            вывод A[1]

иначе
  если ans = A[1] то:
    вывод "не найдено К-ое число"
  все
иначе
  вывод ans

все
все
конец алгоритма.

Задание N5

алг задача 5
начало алгоритма
  вещ A[N;N]
  цел x, y, x0, y0, i, j, N, x1, y1

В задаче нет ограничений на N
Т.к мы не знаем какие числа в таблице, то укажем вещественный тип.

X0=1
Ввод N
Если N<=0 то:
  вывод "некоректное N"
иначе
  y0=N
  x1=0
  y1=0
  Пока x0<>N или y0<N делать:
    нц
    Если A[x0,y0]>0 то:
      x1=x0
      y1=y0
    все
    Если y0-1>0 то:
      y0=y0-1

иначе задача не имеет смысла

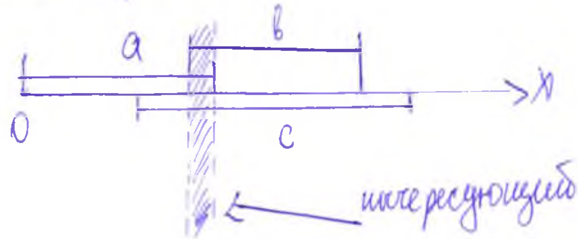
```
иначе
    x0=x0+1
все
x=x0
y=y0
пока x+1<N и y+1<N делать:
нц  x=x+1
    y=y+1
    если A[x,y]>0 то:
        x1=x
        y1=y
    все
кц
кц
если x1=0 и y1=0 то:
    вывод "положительных чисел не встречалось"
иначе
    вывод x1,y1
все
все
конец алгоритма
```

Решаем задачу следующим образом: находясь в ячейке с координатами x0, y0 создаем x=x0, y=y0 и перемещаемся $x=x+1$, $y=y+1$ по таблице пока не дошли до края. После чего делаем y0=y0-1 (если y0>1) или x0=x0+1 (если x0>1). Повторяем эти действия пока не дойдем до A[N,1]. Причем случаи A[N,1] и A[1,N] отдельно считать не нужно т.к. в программе они учтены. Не забываем для каждой ячейки проверять A[i,j]>0? Выводим координаты последнего положит. числа.

Задание №3

Если переформулировать задачу, то имеем: n брёвен длины K, координаты их начала и длину ущелья d. Нужно найти участок, где пересекается (а точнее лежат друг на друге) наибольшее число брёвен.

Рассмотрим координатную прямую OX и отрезки a, b, c на ней



интересующий нас участок (толщина = 3)

Заменим прямую OX на массив A длиной (d+k). Длинной d+k — чтобы потом не было ошибки выхода за границу массива. Теперь если бревно начинается в точке x, то в A[x] ~~[illegible]~~ = A[x]+1, а в A[x+k] = A[x+k]−1. Проходя по массиву устанавливаем значение каждой ячейки, как A[i] = A[i−1] + A[i], где i > 1. Таким образом, там где пересекается больше всего брёвен будет самое большое число (толщина)

алг задача 3

начало алгоритма

```
    цел A[d+k]                      на длину d нет ограничений
    цел d, k, n, i, max
    Ввод n, k, d
    для i от 1 до n:
    нц
    ~~Ввод n, k, d~~  A[i]=0        заполним A нулями
    кц
    Если n<=0 или k<=0 или d<=0 то:   иначе задача не имела бы смысла
       вывод "некорректный ввод"
    иначе
       для i от 1 до n
       нц
         ввод x                      вводим координаты начала i-ого бревна.
```

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

A[x]=A[x]+1
A[x+k]=A[x+k]-1
кц
max=+1
для i от 2 до n
нц
A[i]=A[i]+A[i-1]
Если A[i]>max то:
max=A[i]
все
кц
вывод max
все
конец алгоритма

изначало A - мн заполнили нулями.
т.к. обязательно будет толщина = 1 (1 бревно)

+

Задание №1

Заметим, что те черты, которые проведены по бокам прямоугольного шаблона есть у каждого знака и нет смысла ставить на них датчики. (черта посередине шаблона тоже везде отсутствует) Нет такого, чтобы еще где либо всегда (во всех четырех знаках) была черта или не было, а значит на каждую из них требуются датчики. Всего черт осталось 4 => датчиков тоже 4.

−

Ответ: 4 датчика

Олимпиада школьников «Надежда энергетики»

СОШ № 20

Место проведения

КМ 34-36

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 7311

ФАМИЛИЯ Трофимов

ИМЯ Евгений

ОТЧЕСТВО Эдуардович

Дата рождения 03.02.2002 **Класс:** 11

Предмет Информатика **Этап:** Заключительный

Работа выполнена на 7 **листах** **Дата выполнения работы:** 16.02.20
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ КМ 34-36

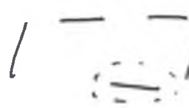
ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

1) Решение. 1) Заметим, что мы можем однозначно определить ~~[illegible]~~ символ

по ~~отключенной~~ непрочерченной правой стороне прямоугольника (пунктир)

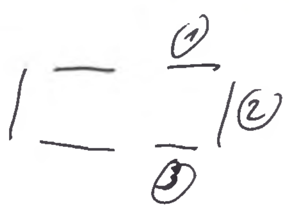
2) Символ  однозначно определяется по непрочерченной правой черте верхней стороны.

т.о мы однозначно определили 2 символа

3) ~~Для~~ ~~определения~~ ~~различ~~ Различить символы  и  мы можем по правой нижней черте.

т.о получаем, что мы можем различить все символы по комбинации 3 черт. Т.е необходимо 3 датчика.

4) Пример: Шаблон:



①,②,③ - датчики;

+, − черта прочерчена или нет

 1 + 2 + 3 −

 1 + 2 + 3 +

 1 + 2 − 3 +

 1 − 2 + 3 +

Видно, что нельзя выбрать 2 датчика из 3 так, чтобы получилось 4 различные комбинации. Следовательно, 3 - минимальное кол-во датчиков.

Ответ: 3. ±

2) var
n - целое;
k - целое;
~~i - целое~~;
kol - целое;
m - целое;
x - целое;
Начало
~~Если k > n~~
ввод (n);
ввод (k);

ввод (m); \ первое число последовательности

~~i := 1~~;

Kol := 1;

пока Kol < K :
нц.
~~i := i+1~~;
ввод(x);
если x > m
начало
m := x;
kol := Kol+1;
конец;
кц;

вывод (m);

конец;

т.о получаем, что ~~числа~~ у нас нет необходимости считывать все числа ячеек и программа работает за линейное время, что гораздо быстрее.

3) var
n, l, bc, x, i, K, max_width - целое;
width [1... l] - массив;

начало
ввод (n); \ Кол-во брёвен

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ КЛЯ 34-36

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

```
ввод(l); \ длина ущелья
ввод(bz); \длина бревна
Для i от 1 до l
   нц
   width[i]:=0;
  кц;
Для i от 1 до n
   Нц
   ввод(x);\ координата начала i-го бревна
   Для k от x до (x+bz)
      Нц
      width[K]:=width[K]+1;
      Кц;
  Кц;
max_width = width[1];
Для i от 1 до l
      Нц.
      Если width[i] > max_width тогда
              max_width:= width[i];
      Кц;
вывод(max_width);
Конец.
```

(+)

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

4) Решение. Пусть $a_1 \dots a_i$ – веса i-х гирь.
$a_1 = 1$ – вес 1-й гири.

2) ~~Заметим, что~~ Докажем, что максимальные значение i-й гири не превосходит значения: $2(a_1 + \dots + a_{i-1}) + 1$.
Пусть это не так. Тогда $a_{i\,max} \geq 2(a_1 + \dots a_{i-1}) + 2$
тогда сумма первых (i − 1) гирь: $a_1 + \dots a_{i-1}$. Но при этом минимальная разность м/у i гирей и $(i-1)$ гирями составляет: $2(a_1 + \dots a_{i-1}) + 2 - (a_1 + a_2 + \dots a_{i-1}) =$
$= a_1 + \dots a_{i-1} + 2$. В таком случае мы не сможем определить вес груза массой $a_1 + \dots a_{i-1} + 1$.

Следовательно, $\boxed{a_{i\,max} = 2(a_1 + \dots + a_{i-1}) + 1}$.

3) ~~Рассмотрим~~ максимальные значения ~~гирь~~ весов гирь: $a_1 = 1$;
~~1; 3; (1+3)·2+1~~ $a_2 = 2 \cdot 1 + 1 = 3$;

$a_3 = (1+3) \cdot 2 + 1 = 9$; $a_4 = (1+3+9) \cdot 2 + 1 = 27$.
Заметим, что $1 + 3 + 9 + 27 = 40$ – выполняется для $Q = 40$.
$a_5 = (1+3+9+27) \cdot 2 + 1 = 81$.

4) (продолжение).
тогда массы первых 5 гирь:
1; 3; 9; 27; 81.
Их сумма: $1+3+9+27+81=40+81=121$.
$121 > 120$.
Следовательно для $Q=120$ кг необходимо пять гирь весами: 1; 3; 9; 27; 81 (кг).
Ответ: 5 гирь; Веса: 1; 3; 9; 27; 81.

5) var

```
i, n, x, y, i; last_x; last_y - целое;
~~a[1..n]~~ a[1...n, 1...n] - ~~целмасс~~ массив;

Начало
ввод(n);
Для i от 1 до n
нц
  Для j от 1 до n
  нц
    ввод(a[i,j]);
  кц;
кц;
Для x от n до 1
нц
  Для y от 1 до (n+1-x)
  нц
```

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

5) (продолжение)

```
   Для i от x до n
     нц
      если a[i, y]>0, тогда
        начало
          last_x:=i;
          last_y:=y;
        конец;
     кц;
   кц;
 кц;
Для y от 2 до n
  нц
  Для x от 1 до (n+1-y)
    нц
     Для
      i от y до n
       нц
       если a[x, i]>0 тогда начало
         last_x :=x;
         last_y:=y;
         конец;
       кц;
       кц;
       кц;
вывод (last_x, last_y);
конец.
```

(+)

Олимпиада школьников «Надежда энергетики»

МЭИ, Москва

Место проведения

BS 88-11

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ Угрюмов

ИМЯ Михаил

ОТЧЕСТВО Андреевич

Дата рождения 13 июня 2004

Класс: 10

Предмет Информатика

Этап: Заключительный

Работа выполнена на 6 листах

Дата выполнения работы: 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: УМА

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача 2.

Пусть, для определённости, груз всегда кладётся на левую чашу весов. Тогда, чтобы получить массу груза, нужно из суммы масс гирь на правой чаше вычесть сумму масс гирь на левой чаше. Таким образом, в данной сумме каждая из гирь умножена на −1, 1 или 0 (на левой чаше, на правой или не задействована). Теперь нетрудно догадаться, что веса ~~грузов~~ гирь являются степенями тройки: 1, 3, 9, 27. У каждой гири 3 состояния, значит возможных конфигураций всего $3^4=81$. Это числа от −40 до 40 включительно (как раз 81 число). И каждому такому числу соответствует уникальная комбинация чисел −1, 1 и 0:

$$-40 = (-1)\cdot 27 + (-1)\cdot 9 + (-1)\cdot 3 + (-1)\cdot 1$$
$$-39 = (-1)\cdot 27 + (-1)\cdot 9 + (-1)\cdot 3 + (0)\cdot 1$$
$$-38 = (-1)\cdot 27 + (-1)\cdot 9 + (-1)\cdot 3 + (1)\cdot 1$$
$$\dots$$
$$0 = 0\cdot 27 + 0\cdot 9 + 0\cdot 3 + 0\cdot 1$$
$$1 = 0\cdot 27 + 0\cdot 9 + 0\cdot 3 + 1\cdot 1$$

+

и так далее.

~~Причём равнее не~~ ~~Грузов~~ Но грузов массы от −40 до 0 не существует. Поэтому как раз на грузы от 1 до 40 и хватает.

Ясно, что при множителях −1, 1 и 0 четырёх гирь не хватит, значит нужно ≥5: 1, 3, 9, 27, 81 ... По аналогии, ими можно закодировать (взвесить) ~~грузы от -3^5 до 3^5~~, ~~то есть~~ $\geq 3^5$ разных чисел (масс грузов): от −121 до 121. То есть любой груз от 1 до 121 кг можно взвесить, если ~~Например~~ есть хотя бы 5 гирь (например, 1, 3, 9, 27, 81).

Ответ: 5 гирь: 1, 3, 9, 27, 81 кг.

Задача 3.

Поскольку никакой закономерности в записанных числах нет, то нужно перебирать их все в порядке, как на схеме, и сравнивать с d и S. После нахождения первого такого числа нужно завершить перебор, для этого понадобится логическая переменная, показывающая, нужно ли продолжать цикл.

Таблицу можно хранить в двумерном массиве, элементами которого будут строки таблицы.

Алгоритм:

1) Объявить N, d, s – целые числа; tab[N][N] - двумерный массив.
2) Считать N, d, s. $N > 0$
3) Для (i от 1 до N с шагом 1):
   1) Для (j от 1 до N с шагом 1):
      1) Объявить k - целое число.
      2) Считать k.
      3) tab[i][j] = k.
   <КОНЕЦ „ДЛЯ">
<КОНЕЦ „ДЛЯ">

Заполнение массива

4) Объявить i = 0, j - целые числа; flag = true - логическая переменная.
5) Пока (i ≤ N и flag = true):
   1) i = i + 1. переходим на следующую строку
   2) Если (i mod 2 = 1): если строка нечетная
      1) j = 0 задаем j = 0
      2) Пока (j < N и flag = true):
         1) j = j + 1.
         2) Если (tab[i][j] ≥ d и tab[i][j] ≤ s):
            1) Вывести „строка = ", i.
            2) Вывести „столбец = ", j.
            3) flag = false.
         <КОНЕЦ „ЕСЛИ">
      <КОНЕЦ „ПОКА">

и идем по строке вправо, ищем нужный элемент

   Иначе: иначе
      1) j = N + 1. задаем j = N+1
      2) Пока (j > 1 и flag = true): и идем

±

Олимпиада школьников «Надежда энергетики»

Вариант: 73101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ BS 88-11

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

1) j = j-1
2) Если (tab[i][j] ≥ d и tab[i][j] ≤ S):
    1) Вывести „строка = ", i.
    2) ~~Строка~~ Вывести „столбец = ", j.
    3) flag = false.

по строке влево, ищем нужный элемент

<КОНЕЦ „ЕСЛИ">
<КОНЕЦ „ПОКА">
<КОНЕЦ „ЕСЛИ, ИНАЧЕ">
<КОНЕЦ „ПОКА">.

~~Конец.~~

~~Таким образом, при нахождении~~

6) Если (flag = true):
    1) Вывести „Нет такого элемента".

<КОНЕЦ „ЕСЛИ">

Конец

Таким образом, при нахождении нужного элемента, алгоритм выводит нужные данные, изменяет значение логической переменной на false и завершает цикл. Если же после завершения цикла логическая переменная не изменилась, значит такого элемента нет.

Задача 5.

Множество Z_m — числа 0, 1, 2, ..., m-2, m-1.
Значит код надёжный, если корни уравнения целые и лежат на промежутке [0; m-1].

Алгоритм:

1) Объявить a, b, c, m, D - целые числа; x_1, x_2 - действительные.
2) Считать a, b, c, m.
3) D = b*b - 4*a*c.
4) Если (D < 0):
    1) Вывод „ненадёжный".

Иначе:

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

1) $x_1 = (-b - sqrt(D))/(2*a)$.

2) $x_2 = (-b + sqrt(D))/(2*a)$. ~~Если хотя бы одно из чисел целое~~

~~3) Если (($x_1 \geq 0$ и $int(x_1) = x_1$) или ($x_2 \geq 0$ и $int(x_2) = x_2$)):~~

3) Если (($x_1 \geq 0$ и $x_1 \leq m-1$ и $int(x_1) = x_1$) или ($x_2 \geq 0$ и $x_2 \leq m-1$ и $int(x_2) = x_2$)):

Если хотя бы одно из чисел $\in [0; m-1]$ и его целая часть равна ему самому ($\Leftrightarrow$ оно целое).

1) Вывод „надёжный".

+

Иначе:

1) Вывод „ненадёжный".

<КОНЕЦ „ЕСЛИ, ИНАЧЕ">

<КОНЕЦ „ЕСЛИ, ИНАЧЕ">.

Конец.

Итак, код надёжный, если у уравнения есть корни и хотя бы один из них целый и $\in [0; m-1]$.

Задача 1.

Для удобства пронумеруем места, где могут быть чёрточки (см. рис.).

Заметим, что во всех четырёх знаках присутствуют чёрточки ①②, ⑥, поэтому они никакой дополнительной информации не несут. Нужно просто запомнить, что они там всегда есть.

А вот чёрточки ③, ④, ⑤ есть не везде, значит для них нужны датчики. Чтобы распознать знак, нужно не менее 3 датчиков.

+

Ответ: 3.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача 4.

1) У каждого слагаемого в знаменателе его номер, а в степени числа x — нечётное число с таким номером. ~~Значит~~: ~~$S_N = \frac{x^{2N-1}}{N}$~~. Но перед ними есть $\underset{\boxed{N\text{-неч.}}}{+}$ или $\underset{\boxed{N\text{-чёт}}}{-}$. $S_N = (-1)^{N+1} \cdot \frac{x^{2N-1}}{N}$

2) Посчитаем количество операций, необходимое для нахождения S_N по формуле из п.1. Это зависит от того, как возводить в степень. Можно просто умножать x на себя $2N-1$ раз, а можно использовать алгоритм быстрого возведения в степень, который работает за $\log_2(2N-1)$ при возведении в степень $2N-1$. Он представляет собой следующую рекурсивную функцию от двух аргументов a и n и возвращает a^n:

$\boxed{\text{Тело функции}}$ power(int a, int n)

```
Если (n=1)
    вернуть a
Иначе если (n=2)
    вернуть a*a
Иначе если (n mod 2 = 1)
    вернуть a* power(a, n-1)
Иначе
    вернуть power(a*a, n/2).
```

Итак, S_N мы находим за $O(\log_2(2N-1))$, так как $(-1)^{N+1}$ занимает несколько операций: если N-чётное, оно равно -1, иначе 1. Деление на N — также $O(1)$.

Если пользоваться другой формулой, то на каждый член мы тратим около 7 операций. Значит на нахождение N-ного — около $7N$ операций. Это дольше, чем формулой 1. Даже если использовать обычное возведение в степень, то 1 формула зачислит около $2N$ операций, всё равно быстрее.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Если же мы пользуемся формулой 1 для нахождения суммы первых N членов, то это займёт $N \cdot \log_2(2N-1)$ операций. ~~А по формуле 2~~ на подсчёт и ещё N на сложение. Это $N + N \cdot \log_2(2N-1)$. А при использовании обычного алгоритма возведения в степень — $N + 2N^2$.

Если мы пользуемся формулой 2 для нахождения суммы первых N членов, мы находим эти члены за $7N$ операций и складываем. Итого — около $8N$.

Для наглядности сделаем таблицу.

	Найти S_N	Найти сумму первых N
формула 1 с медленным алгоритмом	$2N$	$2N^2 + N$
формула 1 с быстрым алгоритмом	$\log_2(2N)$	$N \log_2(2N) + N$
формула 2.	$7N$	$8N$

Как видно, для нахождения одного члена лучше пользоваться формулой 1 в любом случае.
А для нахождения суммы первых N гораздо лучше формула 2.

Ответ: S_N — формула 1 быстрее; сумма первых N — формула 2 быстрее.

±

Олимпиада школьников «Надежда энергетики»

МЭИ город Москва

Место проведения

Ня 67-32

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73111

ФАМИЛИЯ ЦЫГАНКОВ

ИМЯ Николай

ОТЧЕСТВО Викторович

Дата рождения 12.05.2002 **Класс:** 11

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: Цыг

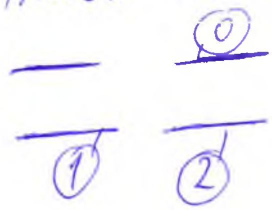
Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

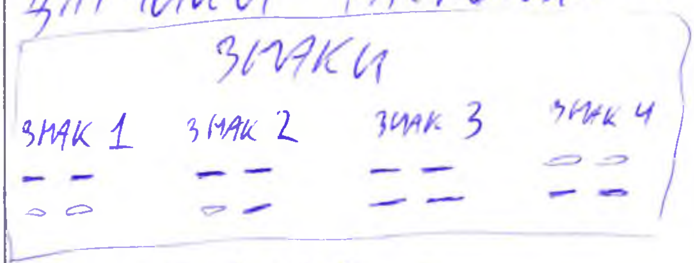
№1

ДЛЯ РАБОТЫ СИСТЕМЫ ПОИСКА ДОСТАТОЧНО 3 ДАТЧИКОВ. ХРАНИТЬ ИНФОРМАЦИЮ С ДАТЧИКОВ БУДЕМ В МАССИВЕ НА 3 ЭЛЕМЕНТА. „1" - ЕСТЬ ЧЕРТОЧКА, „0" - НЕТ ЧЕРТОЧКИ.

```
main{ array int a[3]; bool flag=true;
    for(int i=0; i<3; i++){
        ВВОДИМ С ДАТЧИКА a[i];
        if(i==0){ if(a[0]==0){write("знак 4"); flag=false;}
        if(i==2){ if(a[2]==0){write("знак 1"); flag=false;}
    }
    if(flag){
        if(a[1]==0){write("знак 2");}
        else{write("знак 3");}
    }
}
```

ДАТЧИКИ РАСПОЛОЖЕНЫ ПО ТАКОЙ СХЕМЕ:





ОБОСНОВАНИЕ: • ЕСЛИ НА ДАТЧИКЕ 0 - НЕТ ЧЕРТЫ (ТО ЭТО ОДНОЗНАЧНО ЗНАК 4)

• ЕСЛИ НА ДАТЧИКЕ 2 - НЕТ ЧЕРТЫ (ТО ЭТО ОДНОЗНАЧНО ЗНАК 1)

• В СЛУЧАЕ ~~ОТСУТСТВИЯ~~ ПРИСУТСТВИЯ ЧЕРТОЧЕК НА ДАТЧИКАХ 0 и 2 ОТВЕТ ОПРЕДЕЛЯЕТ ДАТЧИК 1 ЕСЛИ НА НЕМ ЕСТЬ ЧЕРТА ТО ЭТО ОДНОЗНАЧНО ЗНАК 3, ЕСЛИ НЕТ ТО ЗНАК 2

ОТВЕТ: 3 ДАТЧИКА

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№2

```
main{int n, k;
    read(n); int a[n]; int ite=0; int min; int ind;
    read(k); if(k>n){write("k не может быть больше n");}
    for(int i=0;i<n;i++){
        read(a[i]);                      //ввод массива
    }
        min=a[0];
    for(int i=0;i<n;
    while (true){
        min=a[ite]; ind=ite;
        for(int i=(ite+1)<n;i++){
        for(int i=ite+1;i<n;i++){
            if(a[i]<min){ind=i;min=a[i]}
        } if(ite+1==k){write(a[ite]
        if(min<a[ite]){a[ind]=a[ite];}
        ite++;
        if(ite==k){write(a[ite-1]);break;} //ответ
    }
```

± ±

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-32

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

N3

```
main{ int n ; reat (n); int[n] a; int[n] y;
    for(int i=0; i<n; i++){
        reat( a[i];)   // ввод начал расположения брёвен
        y[i] = 0;
    }
    for(int i=0; i<n; i++{ y[a[i]]++; }
    int otver = y[0];
    for(int i=0; i<n; i++){if (y[i] > otvet ){otvet=y[i];}
    write (otvet);
}
```

+

N4

ТАК КАК для Q=40 нам минимально требуется 4 гири

оценка сверху

то для Q=120 будет ~~достаточно 6 гирь:~~ 40 и 40 и набор для Q 40 так как двумя задачу к Q=40 гирьками по 40 можно сводить логично предположить что:

1 гирькой по 20 можно свести Q=40 к Q=20

1 гирькой по 10 свести Q=20 к Q=10

1 гирькой по 5 свести Q 10 к Q 5

Q_5 = 5 и 3

+

Ответ: 6 гирь : 20; 10; 5; 3; 40; 40

Олимпиада школьников «Надежда энергетики»

Вариант: 73111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ МЯ 67-32

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№5

```
                                     int ~~otvet y=0, otvet x=0~~;
main{   int n; read(n); int[n] a ;int j; int startX, startY;
  for(int j=0; j<n; j++) {
  for(int i=0; i<n; i++){
     read ( a[j,i] );                  // входные данные
  }}
 otvet x = 0; otvet y = n-1; ~~min=999999~~                    
 for(int i = 1; i < n·2-1; i++){          // кол-во диагоналей
   j = 2n-1; if(i ≤ n){j = i;}           // "j" длина диагонали
   ~~startX = ...~~
   ~~startY = ...~~
   startX = n-i; if(i≥n){startx = 0;}
   starty = 0; if(i>n){starty = i-n;}
   for( ; j>0; j--){                    // проходим массив по пути
     ~~if(a[start~~
     if(a[start Y, start X] ≥ 0){otvet y=start y;
                                   otvet x= start x;}
     start y ++;
     start x ++;
   }
 }
 write ("строка" +(otvet y+1));
 write ("столбец"+(otvet x+1));
}
```

+

Олимпиада школьников «Надежда энергетики»

г. Калининград

Место проведения

ФS 60-11

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 73991

ФАМИЛИЯ ШИБАЕВ

ИМЯ АЛЕКСАНДР

ОТЧЕСТВО Михайлович

Дата рождения 29.10.2004 **Класс:** 9

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 **листах** **Дата выполнения работы:** 16.02.2020
(число, месяц, год)

Подпись участника олимпиады: ____________

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Задача №1.

Очевидно, что 1 датчика не хватит, т.к. у нас всегда будет 2 или 3 возможных варианта расшифровки. Докажем, что при 2 датчиках можно могло определить

……. - это пустая сторона

v - значит стоит датчик

v1 - первый датчик

v2 - второй датчик

±

Если v1 и v2 - „говорят" что карта прогрета, то это ③ символ, если оба говорят „нет", то это ①, а если ~~нет~~ V1 - „нет" а V2 - „да", то это ②.

Задача №2.

1 3 9 27

$1+3+9+27=40$

$1=1$
$2+1=3$
$3=3$
$4=3+1$
$5+1+3=9$
$6+3=9$
$7+3=9+1$
$8+1=9$
$9=9$
$10=9+1$
$11+1=9+3$
$12=9+3$
$13=9+3+1$
~~14+1+3+9=27~~ $14+1+3+9=27$
$15+3+9=27$
$16+9+3=27+1$
$17+1+9=27$
$18+9=27$

$19+9=27+1$
$20+9+1=27+3$
$21+9=27+3$
$22+9=27+3+1$
$23+3+1=27$
$24+3=27$
$25+3=27+1$
$26+1=27$
$27=27$
$28=27+1$
$29+1=27+3$
$30=27+3$
$31=27+3+1$
$32+1+3=27+9$
$33+3=27+9$
$34+3=27+9+1$
$35+1=27+9$
$36=27+9$
$37=27+9+1$
~~38=27+~~ $38+1=27+9+3$

$39=27+9+3$
$40=27+9+3+1$

+

Олимпиада школьников «Надежда энергетики»

Вариант: 73991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ ФS 60-11

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

Задача №3

```
#include<bits/stdc++.h>
using namespace std;
long long f(int n){
    long long a=1;
    for(int i=1; i<=n; i++){
        a=a·10;
    }
    return a;
}
int check(int x, int y, int z){
    long long a=0;
    a += 20·f(12);
    a+= x·f(11);
    a+= z·f(10);
    a+= 9·f(9);
    a+= y·f(8);
    a+= x·f(7);
    a+= 2019·f(3);
    a+= x·f(2);
    a+= y·f(1);
    a+=1;
    if(a% (10y+x) ==0){
        return 1;
    }
    return 0;
}
int main(){
    long long k=0;
    for(int x=0; x<=9; x++){
        for(int y=0; y<=9; y++){
            for(int z=0; z<=9; z++){
                k += check(x, y, z);
            }
        }
    }
    cout << k;
    return 0;
}
```

Алг. + Адр.

±

Задача №5.

Т.к. я не совсем понял условие задачи, то я приведу 2 решения: первое: где сам корень должен быть в множестве Z_m, а во втором: где все вычисления производятся по модулю m и корень должен находится в Z_m.

первое:

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int a,b,c,m;
    cin>>a>>b>>c>>m; //вводим a,b,c,m.
    if(b*b-4*a*c<0){        // если D<0, то выводим, что код ненадежен.
        cout<<"NO";
        return 0;
    }
    int k=sqrt(b*b-4*a*c);
    if(k*k==b*b-4*a*c){
        if(((-1)*b+k>=0 && ((-1)*b+k)%(2*a)==0)    // & - это амперсант
          ||((-1)*b-k>=0 && ((-1)*b-k)%(2*a)==0)){
            cout<<"Yes";
            return 0;

        }
    }
    cout<<"NO";
    return 0;
}.
```

второе.

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

```
#include <bits/stdc++.h>
using namespace std;
int md (int a, int m){
    if (a<0){
        return m - (a%m);
    }
    return a%m;
}
int main(){
    int a,b,c,m;
    cin >>a>>b>>c>>m;
    int d= md(b·b,m) + md((-4)·a·c,m);
    d= md(d,m);
    int k= sqrt(d);
    if (k·k == d){
        if ( md(k-b, m) % md(2·a,m) ==0
            || md ((-1)·k-b,m) % md (2·a,m) ==0){
            ~~cout<<"Ye~~
            cout<< "Yes";
            return 0;
        }
    }
    cout << "NO";
    return 0;
}
```

±

Задача №4.

Если в самом начале все красные стоят самыми последними, а все зеленые - самыми первыми, то надо будет сделать ходов:

все красные в начало: $2\cdot(N-x)\cdot x$, x - число красных.

Теперь можем считать, что ряд длиной $N-x$.

переместим все зеленые в конец: не более $2\cdot(N-x-y)\cdot y$ ходов (y число зеленых).

~~Всего~~: Всего: $2(N-x)\cdot x + 2\cdot(N-x-y)\cdot y < 4N$

$x+y \le N$

$2Nx - 2x^2 + 2Ny - 2xy - 2y^2 \le 4(x+y)$

$2N(x+y) - (x+y)^2 - (x^2+y^2) \le 4(x+y) \quad | : x+y$

$2N - (x+y) - P \le 4$

−

Олимпиада школьников «Надежда энергетики»

СФМЭИ

Место проведения

NC 75-69

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 73101

ФАМИЛИЯ Яцук

ИМЯ Александр

ОТЧЕСТВО Владимирович

Дата рождения 09.06.2004

Класс: 10

Предмет информатика

Этап: Заключительный

Работа выполнена на 3 листах

Дата выполнения работы: 16 февраля 2020
(число, месяц, год)

Подпись участника олимпиады:

Впишите свою фамилию имя и отчество печатными буквами, дату рождения, класс, название предмета, этапа Олимпиады, общее количество листов, на которых выполнена работа и дату выполнения работы.

Олимпиада школьников «Надежда энергетики»

Вариант: 13101 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ НС 75-69

ВНИМАНИЕ! Проверяется только то, что записано с этой стороны листа в рамке справа

№1

рассмотрим позиции которые даны

Эти положения имеют 3 общих положения а остальные изменяются. Исходя из это можно купить 3 датчика для положений а положения считать всегда зажженными.

~~Ответ: 3~~ Так же заметим, если правый индикатор и правый нижний представить как 1 или 0 то по ним можно однозначно ~~задать~~ узнать какое это положение.

Например: = 01 = 11 = 10 = 00

Ответ: 2

+

№2

Для решения поставленной задачи воспользуемся 3-ой системой счисления. Пусть 0 ~~означает~~ в 3-ой записи значит что мы не берем эту гирю 1 что берем, и 2 что мы берем следующую по наименьшему и вычитаем эту.

+

Пример: $W = 89_{10} = 10022_3$. Для удобства над каждой цифрой напишем нужную степень 3-и ~~10022~~ $\overset{81}{1}\,\overset{27}{0}\,\overset{9}{0}\,\overset{3}{2}\,\overset{1}{2} = 81+9-3+3-1$ произведя все сокращения получим $81+9-1=89$. Для $W \leq 90$ можно использовать лишь 5 гирь максимум 81, 27, 9, 3, 1 т.к. их сумма > 90

Ответ: 5 гирь (81, 27, 9, 3, 1)

№3

N>0

Введём входную таблицу $N \times N$ в массив a[N][N]

переберём номер ~~столбца~~ строки i от 1 до N

если i чёт то

переберём номер столбца j от N до 1

если a[i][j] ≥ d и a[i][j] ≤ S то

выводим i, j; завершаем программу

иначе переберём номер столбца j от 1 до N

если a[i][j] ≥ d и a[i][j] ≤ S то

выводим i, j; завершаем программу.

±

№4

Приведём интуитивную формулу $S(x) = x - \frac{x^3}{2} + \frac{x^5}{3} - \frac{x^7}{4} + \ldots + (-1)^{n+1}\frac{x^{2n-1}}{n}$

докажем: $n=1$ $S(x) = (-1)^2\frac{x^{2-1}}{1} = x$

$n=2$ $S(x) = x + (-1)^3\frac{x^{2\cdot 2-1}}{2} = x - \frac{x^3}{2}$

$n=3$ $S(x) = x - \frac{x^3}{2} + (-1)^4\frac{x^{6-1}}{3} = x - \frac{x^3}{2} + \frac{x^5}{3}$

Количество операций для каждого элемента прогрессии будет $3n+3$ значит для подсчёта всех арифм. операций потребуется $\approx n + \frac{(3n+3)^2}{2} = n + 4,5n^2 + 4,5 + 9n = \boxed{4,5n^2 + 10n + 4,5}$

значит для ~~выполнения~~ подсчёта арифм. действий выражения $\frac{S_{n+1}}{S_n} \approx 4,5n^2 + 10n + 4,5 + 4,5(n+1)^2 + 10n + 10 + 4,5 + 1 =$

$= 4,5n^2 + 10n + 4,5 + 4,5n^2 + 4,5 + 9n + 10n + 10 + 4,5 + 1 = 9n^2 + 29n + 24,5$ действий

а для выражения $-x^2\frac{N}{N+1}$ потребуется 5 действий.

+

№ 5

Зная a, b, c и m, достаточно проверить на пр-[illegible] корни.

читаем a b c m

Заведем число D равное b^2-4ac

если $D<0$ ответа не существует, то есть Вася заперт ненадёжно

Заведём $X1=(-b+\sqrt{D})/2\cdot a$

Заведём $X2=(-b-\sqrt{D})/2\cdot a$

если $(X1\geq 0$ и $X1<m)$ или $(X2\geq 0$ и $X2<m)$ то Вася заперт надёжно

иначе Вася заперт ненадёжно

-