Олимпиада школьников «Надежда энергетики»

Москва

Место проведения

RV 64 - 99

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ АРИСОВА

ИМЯ ЕЛИЗАВЕТА

ОТЧЕСТВО ИВАНОВНА

Дата рождения 26.09.2004 **Класс:** 11

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 05 **листах** **Дата выполнения работы:** 20.02.2022
(число, месяц, год)

Подпись участника олимпиады: АИ

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»

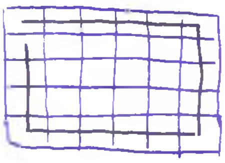
Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RU 67 - 99

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

Задача №1

Пусть нумерование элементов таблицы начинается с нуля, а значит заполнение следует начинать с элемента [0;0]



Заметим, что при проходе одного круга в таблице мы окажемся в элементе с индексом [1,1], и нам придётся повторить задачу для таблицы размером (M-2)*(N-2) с первым элементом [1,1]

Пусть ~~в~~ список posl - последовательность чисел, которую нужно внести в таблицу.

```
is = 0          # координаты левого верхнего угла
js = 0
Flag=0          # Flag - номер элемента послед. таблицы posl
for k in range (min(N, M)//2):
    j = js
    i = is
    while j< M-js-1:
        a[i;j]= posl[flag]
        Flag+=1
        j+=1
    while i< N-is-1:
        a[i; j] = posl[Flag]
        flag+=1
        i+=1
    while j>0+js:
        a[i;j] = posl[flag]
        Flag+=1
        j-=1
    while i>0+is
        a[i;j] = posl[flag]
        flag+=1
        i-=1
    is+=1
    js+=1
if M%2!=0:
    a[i][j]= posl[flag]
```

Для каждой из 4-х сторон таблицы реализуется алгоритм: если таблица не закончилась, сдвинуться на 1 элемент, а если закончилась, начинаем идти по следующей стенке.

Заметим, что после выполнения цикла for, один элемент таблицы не будет заполнен, если M-нечетное. Ему значение присваивается в конце программы

Задача 2

Пусть массив с координатами точек называется m.

Для начала приведем функцию, считающую периметр точек с индексами a; b; c:

```
def perimetr (a, b, c):
    ab = sqrt((m[a][0]-m[b][0])^2 + (m[a][1]-m[b][1])^2 +
        + (m[a][2]-m[b][2])^2)
```

Аналогично посчитаем значения переменных bc и ac - длин сторон треугольника

```
    if ab != 0 and bc != 0 and ac != 0:
        return ab+bc+ac
    else:
        return 0
```

Проверка на совпадающие точки.

Еще бы добавить проверку на то, что они не на 1 прямой!

Алгоритм для перебора точек в массиве таким образом, чтобы не попадались повторяющиеся наборы точек или одна точка не находилась в наборе дважды:

```
maxp = 0
for a in range (0; N-2):
  for b in range (a+1; N-1):
    for c in range (b+1; N):
      if perimetr (a; b, c) > maxp:
        maxp = perimetr (a, b, c)
        answer = [a, b, c]
print (answer)
```

Получается, что для каждого набора точек a < b < c => индексы никогда не совпадут и не повторятся

Программа для каждого набора точек находит периметр треугольника и сравнивает его с максимальным на текущий момент. Если периметр больше, она создаёт список answer с индексами точек.

В качестве ответа она выведет индексы точек - вершин Δ с наибольшим периметром.

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RU 67-99

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

Задача 4

~~Беспр~~

Дано:

I последовательность – 8 чисел, в каждом 16 разрядов

II последовательность – 16 чисел, в каждом 8 разрядов

Решение:

1) Число из I посл имеет 16-1=15 разрядов без знакового
Число из II посл. имеет 8-1=7 разрядов без знакового
2) Максимальное число в I посл $= 2^{15}$ по модулю
Максимальное число в II посл $= 2^{7}$ по модулю
3) Значит, максимальное произв. чисел $= 2^{22}$ по модулю
4) Для его хранения необходимо 22 разряда + 1 знаковый разряд => 23 разряда.
5) Значит, в памяти нужно хранить числа с 23 разрядами (произведения 2-х чисел)
6) При попарном умножении получится $8 \cdot 16 = 2^7$ чисел (~~всего~~ для 1 пары последовательностей)
7) ~~[illegible]~~

Максимальная сумма ~~[illegible]~~ попарных произведений равна $2^{22} \cdot 2^{7} = 2^{29}$.
На её хранение необходимо $29 + 1$ (знаковый) $= 30$ разрядов

8) После того, как посчитана сумма попарные произведения можно удалять из памяти.

9) Для одной пары посл. необходимо
- 2^{7} чисел с 23 разрядами (произведения)
- 1 число с 30 разрядами (сумма)

10) Для всех пар посл. необходимо выделить место для:
- 2^{7} чисел с 23 разрядами (произведения могут не храниться в памяти после подсчёта сумм)
- 2^{10} чисел с 30 разрядами (суммы)

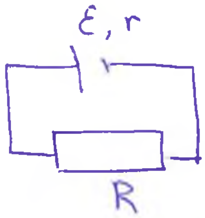
Ответ: 2^{7} чисел с 23 разрядами и 2^{10} чисел с 30 разрядами

×10

Задача 5

Дано:
$\mathcal{E}$ - const
r - const
известны I и U

Найти
$\mathcal{E}$ - ?
r - ?



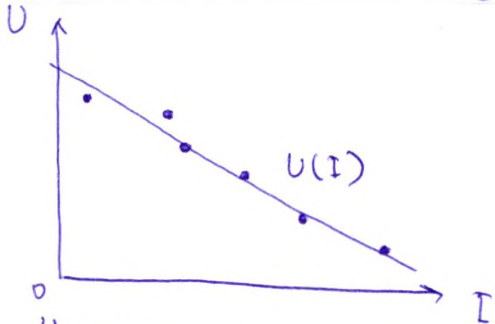
1) $I = \frac{\mathcal{E}}{r+R} \Rightarrow \mathcal{E} = I\cdot r + I\cdot R$ ✓

2) $I = \frac{U}{R} \Rightarrow U = I\cdot R$ ✓

3) $\mathcal{E} = I\cdot r + U$

4) для каждого эксперимента известны значения I и U, а $\mathcal{E}$ и r – не изменяются. Значит, переписав уравнение:
$U = -I\cdot r + \mathcal{E}$, мы получим линейную функцию U(I) вида $y = kx + m$, где $-r = k$
$\mathcal{E} = m$

5)



На координатной плоскости отметим 2N точек, у которых координата x – значение I для данного эксперимента, а координата y – значение U.

Через эти точки проведем прямую, это будет график функции U(I)

6) По графику можно определить:
· U(0) – значение U при I = 0
· I_0 – значение I при U = 0

7) $\mathcal{E} = U(0)$
$r = \frac{\mathcal{E} - U}{I} = \frac{\mathcal{E}}{I_0}$

Ответ: $\mathcal{E} = U(0)$ $r = \frac{\mathcal{E}}{I_0}$

+9

Нужен алгоритм без участия оператора – как найти I_0 не графически?

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

Задача №3

Пусть a – исходная матрица,
b – новая усредненная матрица

```
for i in range (N):
  for j in range (N):
    if i==0 or j==0 or i==N-1 or j==N-1 :
      b[i][j]= a[i][j]
    else:
      summ=0
      for k in range (i-1, i+2):
        for m in range (j-1, j+2):
          summ += a[k][m]
      b[i][j] = summ/9
```

Крайние элементы таблицы невозможно усреднить, поэтому они остаются прежними

+10

Олимпиада школьников «Надежда энергетики»

№ группы	Место проведения
19F-01	Дистанционно, с использованием ВКС

WZ73-69

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 37991

ФАМИЛИЯ Всеводина

ИМЯ Дарьяна

ОТЧЕСТВО Евгеньевна

Дата рождения 09.06.2006 Класс: 9

Предмет Информатика Этап: Заключительный

Работа выполнена на 2 листах Дата выполнения работы: 20.02.2022
(число, месяц, год)

Подпись участника олимпиады: [illegible]

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ WZ73-69

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

+10

Задача 1

$5E_n = ***_{10} = * * 0_8$

Найти минимальное n

$5E_n = E \cdot n^0 + 5 \cdot n^1 = E + 5 \cdot n_{10}$

"E" соответствует 14 в десятичной системе счисления

Значит система счисления не ниже 14ой

Если $n = 18$:

$E_{18} + 5 \cdot 18 = 14 \cdot 18^0 + 5 \cdot 18 = 14 + 90 = 104_{10}$ - 3 разряда

$104_{10} = 150_8$ - заканчивается нулем

Условия задачи выполнены, значит минимальное $n = 18$

Ответ: 18

Задача 2

+10

```
int a[1000];
int M, N;
cin >> M >> N;
for (int i=0; i<N; i++)
{ cin>> a[i]; //считываем длина кусков провода
while (a[i]> a[i-1])
{ swap (a[i-1], a[i]); }
int S=0;
int Kol=0; //переменные для подсчета кол-ва кусков провода и возможной длина
while (M<S)
{ S=S+a[i];
  Kol++;
}
cout<<Kol //выводим кол-во кусков провода
```

Олимпиада школьников «Надежда энергетики»

Вариант: 35391

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ WZ73-69

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

Задача 3:

```
int a[10000];
int N, sum=0, sredn=0;
cin>>N;
for (int i=0, i<N, i++)
{ cin>>a[i] //считываем значения
  sum=sum+a[i];
}
  sredn= sum/N; //находим среднее значение
for (int i=0, i<N, i++)
{ if (sredn*2 < a[i])
      cout<<a[i]; //выводим результаты
}
```

+6

найти отклонения и среднее отклонение

Задача 4:

Первая последовательность: 8 чисел по 16 разрядов

Всего: 8·16 = 128 разрядов

Вторая последовательность 8 чисел по 8 разрядов

Всего 8·8 = 64 разряда

При перемножении первой последовательности на вторую получим 128+63=191 разрядное число

Мы считываем 1024 пары последовательностей, значит мы получим 1024 числа с разрядностью 191

+10

Ответ 1024 числа

Разрядность 191

Задача 5

```
a[N]; //отсортированный массив по убыванию
for (int i=0, i<N, i++)
{ a[i] = a[N-1-i]; //сортируем массив по возрастанию
}
```

+2

почему?

Лист 02 из 02

Вари...

Олимпиада школьников «Надежда энергетики»

I11F01	Дистанционно, с использованием ВКС
№ группы	Место проведения

RO51-38

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ Лисенко

ИМЯ Алёна

ОТЧЕСТВО Леонидовна

Дата рождения 27.05.2004 Класс: 11

Предмет Информатика Этап: Заключительный

Работа выполнена на 4 листах Дата выполнения работы: 20.02.22
(число, месяц, год)

Подпись участника олимпиады:

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37/11

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-38

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

№1 Чтобы решить задачу будем в цикле идти ~~по в таком порядке~~: вправо, вниз, влево, вверх, при этом заполняя на какой клетке мы остановились

- Язык Python -

```
m,n= map(int,input().split()) # ввод m и n
num=[]
for i in range(m+1):
    h=[]
    for j in range(n+1):
        h.append(0)
    num.append(h)
# заполнили таблицу нулями
now=1 # текущее натуральное число
lasti=1 # текущая строка
lastj=1 # текущий столбец
while now<n*m:
    ~~if now!=1: # когда только запускаем цикл, прибавлять к строке~~
    ~~lastj+=1 не надо, иначе т.к. эту клетку уже отработали +1~~
    while lastj<n and num[lasti][lastj+1]==0:
        num[lasti][lastj]=now   # заполняем числом, идем
        now+=1                  вправо пока не достигнем
        lastj+=1                конца матрицы или уже заполненой клетки
    while lasti<m and num[lasti+1][lastj]==0:
        num[lasti][lastj]=now   # заполняем числом,
        now+=1                  идем вниз аналогично
        lasti+=1
    while lastj>1 and num[lasti][lastj-1]==0:
        num[lasti][lastj]=now  # заполняем числом, идя
        now+=1                 влево аналогично
        lastj-=1
    while lasti>1 and num[lasti-1][lastj]==0:
        num[lasti][lastj]=now   # заполняем, идя
        now+=1                  вверх аналогично
        lasti-=1

# вывод:
for i in range(1,m+1):
    for j in range(1,n+1):
        print(num[i][j],end=' ')
    print()
```

+10

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

№2

```
def get_len(x,y,z,x1,y1,z1): # ф-ия нахождения рас-ния м/у 2-мя точками
    return((x-x1)**2+(y-y1)**2+(z-z1)**2)**0.5
def get_per(a,b,c): # нахождение периметра Δ abc, a,b и c - кортежи координат точек
    len1=get_len(a[0],a[1],a[2],b[0],b[1],b[2])
    len2=get_len(b[0],b[1],b[2],c[0],c[1],c[2])
    len3=get_len(c[0],c[1],c[2],a[0],a[1],a[2])
    return len1+len2+len3
x2,y2,z2=0,0,0 # здесь будем хранить индексы точек, Δ с наибольшим периметром
ans=-1 # текущий максимальный периметр
n=int(input())
~~for i in ra~~
points=[]
for i in range(n):
    x,y,z=map(int,input().split())
    points.append((x,y,z))
#Ввели все точки
# чтобы решить задачу, переберем все возможные тройки точек
  и выберем ту, у которой периметр максимальный
for i in range(n):
    for j in range(i+1,n):
        for k in range(j+1,n):
            ~~ans=max(ans,get_per(points[i],points[j],points[k])~~
~~print(ans)~~ if ans< get_per(points[i],points[j],points[k]):
                x2,y2,z2=points[i],points[j],points[k]
                ans=get_per(points[i],points[j],points[k]):
print(x2,y2,z2,ans)
```

-выводит кортежи координат точек с наибольшим периметром и периметр Δ, который они образуют

~~Периметр каждого~~ Длину каждого отрезка вычисляли по формуле: $d=\sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2}$, где x,x_1,y,y_1,z,z_1 - координаты точек

+7

Нет проверки существования Δ

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

$\boxed{№3}$ Решение: будем в цикле перебирать каждый элемент исходной матрицы и для него считать сумму соседних элементов в пределах 3×3, дальше в новую матрицу записывать на место текущего элемента полученную сумму, разделенную на 9 (по формуле среднего арифметического)

```
~~n=int(input~~
n=int(input())
m=[] - исходная матрица
for i in range(m):
    m.append(list(map(int, input().split())))
new_m=m - новая матрица
for i in range(n):
    for j in range(m):
        summ=0 - # сумма соседних элементов
        count=0 # кол-во элементов в сумме (т.к. если обрабатываем
                 один из крайних элементов, вокруг него менее 9 эл.)
        for i1 in range(max(0,i-1), min(n-1,i+1)+1):
            for j1 in range(max(0,j-1), min((n-1),j+1)+1):
                summ+= m[i1][j1]
                count += 1
        new_m = summ//count
        [индексы ???
print(new_m, sep='/n') #вывод
```

(+8)

$\boxed{№4}$ Для каждой пары ~~последовательностей~~ будет 1 итоговая сумма (сумма всех попарных произведений членов 2-ух последовательностей) ~~...~~ В сумме будет 16×8 чисел, 16×8=128 чисел

Т.к. число из 1-ой посл. 16-разрядное со знаком, то без знака имеет 15 разрядов, аналогично из 2-ой последов-ти все числа 7-ми разрядные (без знака). значит произведение числа из 1-ой и 2-ой будет иметь: 15+7-1+1 = 22 разряда (с учетом знака).

См. страницу

Олимпиада школьников «Надежда энергетики»

Вариант: 37111 ШИФР, НЕ ЗАПОЛНЯТЬ ⇨ RO51-38

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

Итого в итоговую сумму каждой пары последовательностей пойдет 128 чисел по 22 разряда, значит итоговая сумма будет иметь 22+8=30 разрядов

Для 1024 пар последовательностей будет храниться 1024 итоговых суммы по 30 разрядов в каждой.

10

№5 По формуле для нахождения ЭДС: $\mathcal{E} = U + Ir$, где U - напряжение, I - сила тока, r - внутреннее сопротивление

Т.к. ЭДС постоянный для всех измерений, где измеряется U_i и I_i, можно составить равенство $U_i + I_i r_i = U_j + I_j r_j$, при этом r - внутреннее сопротивление всегда одно, значит равенство принимает вид: $U_i + I_i r = U_j + I_j r \Rightarrow r = \frac{U_i - U_j}{I_j - I_i}$ ✓

Чтобы найти среднее внутреннее сопротивление будем перебирать все результаты экспериментов и добавлять полученное для них r в массив, а далее найдем среди всех среднее. Пусть результаты экспериментов будут вводиться в формате "U_i I_i"

```
n = int(input())
res = [ ]
for i in range(n):
    U, I = map(int, input().split())
    res.append((U, I))
sr = [] - массив для всех r
for i in range(n):
    for j in range(i+1, 2*n):
        sr.append((res[i][0] - res[j][0]) / (res[j][1] - res[i][1]))
r = sum(sr)/len(sr) - # находим среднее среди всех r
E = res[0][0] + r * res[0][1] - # по формуле находим ЭДС
print(r, E)
```

10

Лист 04 из 04

Олимпиада школьников «Надежда энергетики»

Москва

Место проведения

RV 67-41

шифр

← **Не заполнять**
Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ МАКАРЕВИЧ

ИМЯ МАРИЯ

ОТЧЕСТВО ДЕНИСОВНА

Дата рождения 08.02.2004 **Класс:** 11

Предмет ИНФОРМАТИКА **Этап:** ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 04 **листах** **Дата выполнения работы:** 20.02.2022
(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: 37111

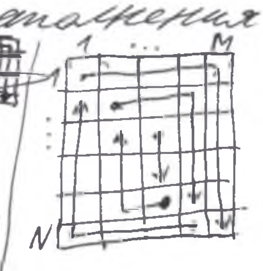
ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RU 67 -41

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

1 ~~Создадим глобальный список r разм. n, каждый эл~~

Считаем введенные пользователем значения M и N, проверим, выполняются ли неравенства N>0 и M>0; если нет, то выведем сообщение: "Некорректные данные" и завершим программу. Иначе: создадим ~~...~~ глобальный список r размером N, каждый элемент которого является списком размера M, заполненным нулями. Нумерация в списке начинается с 1.

Создадим две рекурсивные функции для заполнения полученной таблицы „галочками":



1) Ф-я rec1 принимает аргументы n, m, a, x, y. Сначала эта ф-я проверяет, выполн. ли одно из равенств: n=0; m=0, и если да, то она завершает работу. Иначе она выполн. следующие действия:

цикл для переменной i от y включительно до (y+m-1) включительно с шагом 1.

В этом цикле: r[x][i]:=a

a:=a+1

После завершения этого цикла выполн. вход в другой цикл для i от (x+1) (т.е. от x невключит.) до (x+n-1) включит. с шагом 1.

В цикле: r[i][y+m-1]:=a

a:=a+1

После завершения цикла ф-я вызывает ф-ю rec2 с аргументами n-1, m-1, a, x+1, y (т.е. rec2(n-1, m-1, a, x+1, y))

2) Ф-я rec2 принимает аргументы n, m, a, x, y. Сначала она выполн. проверку n=0, m=0 аналог. ф-и rec1, и если n≠0, m≠0, то выпол. следующие действия:

цикл для i от (y+m-1) включ. до y включ. с шагом -1.

В цикле: r[x+n-1][i]:=a

a:=a+1

Конец цикла

Цикл для i от (x+n-2) (т.е. (x+n-1) невключ.) до x включ. с шагом -1.

В цикле: r[i][y]:=a

a:=a+1

Конец цикла

Вызов ф-и rec1(n-1, m-1, a, x, y+1).

неоптимально. +9

~~3) Вызов ф-и rec1~~ В основном теле программы: ~~вызов ф-и~~ rec1(N, M, 1, 1, 1).

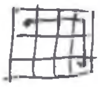
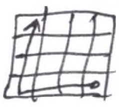
После завершения работы программы в r будет лежать

Олимпиада школьников «Надежда энергетики»

Вариант: 37111 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RV 67 - 41

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

искомая матрица.

Примечание действие rec1:  ; rec2:  ; x, y — индексы левой верхней клетки той части таблицы, с которой работает программа в данный момент.

2) Пусть массив с коорд. точек называется r, нумерация в нем начинается с 0.

Создадим список ds длиной N, каждый элемент которого является списком, состоящим из двух списков длины 2, заполн. нулями. Т.е. эл-ты ds выглядят так: [[0, 0], [0,0]].

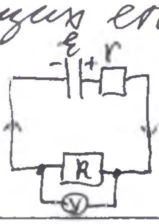
Начало

Цикл для i от 0 включ. до N невключ.:
    Цикл для j от i невключ. до N невключ.:
        считаем в перемен. x_1, y_1, z_1 знач. эл-тов r[i] (т.е. x_1, y_1, z_1 = r[i])
        аналогично x_2, y_2, z_2 = r[j]
        в перемен. d положим расстояние между точками с индексами i и j, вычисл. по формуле $d := \sqrt{(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2}$
        если d > ds[i][0][0]:
            ds[i][0] := [d, j]
        иначе если d > ds[i][1][0]:
            ds[i][1] := [d, j]
        конец если
        если d > ds[j][0][0]:
            ds[j][0] := [d, i]
        иначе если d > ds[j][1][0]:
            ds[j][1] := [d, i]
        конец если
    конец цикла для j
конец цикла для i
maxi := 0
ans := [ ] #пустой список
Цикл для i от 0 включ. до N невключ.:
    запишем в перемен. d_1 расст. между точками с индексами ds[i][0][1] и ds[i][1][1], вычисл. по той же формуле, что и раньше: $d_1 := \sqrt{(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2}$, где x, y, z — коорд. вышеупомянутых точек.
    d := ds[0][0] + ds[1][0] + d_1
    если d > maxi:
        maxi := d
        ans := [i, ds[i][0][1], ds[i][1][1]]
    конец если
Конец цикла
Конец

После заверш. программы в перемен. maxi будет лежать знач. наибольшего периметра, а в перемен. ans — индексы составляющих его точек в начальном массиве.

Нет проверки ∃ △ — точки могут лежать на одной прямой

F7

5)  $U = IR = \mathcal{E} - Ir \Rightarrow \mathcal{E} = Ir + U$

Пусть данные о результатах экспериментов хранятся в массиве int (индексация с 0); 0-й эл-т каждого

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RV 67 - 41

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

элемента этого массива равен знач. I, получ. в эксперименте, а 1-й эл-т равен U; E := ~~1~~ -1 - эдс, r := ~~0~~ -1 - сопротивл.

```
prevI := inf[0][0]
prevU := inf[0][1]
Цикл для i от 1 включ. до 2N невключ.:
    I := inf[i][0]
    U := inf[i][1]
    r1 := (prevU - U) / (I - prevI)
```

$\left(т.к. \begin{cases} \mathcal{E} = prevI \cdot r + prevU \\ \mathcal{E} = I \cdot r + U \end{cases} \Rightarrow r(I - prevI) = prevU - U\right)$

```
    если r != -1:
        r := (r + r1) / 2 #средн. знач.; деление не целочисленное
    иначе:
        r := r1
    конец если
    E1 := I·r + U
    если E != -1:
        E := (E + E1) / 2
    иначе:
        E := E1
    конец если
Конец цикла
Вывод E, r
Конец
```

x10

3) Пусть матрица назыв. r; индексац. с 0. Усреднение по значениям не может быть выполнено для эл-тов ~~для строк~~ крайних строк и столбцов, т.к. у них недостаточно соседних значений.

Надо как-то выходить из положения!

Цикл для i от 0 включ. до (N-2) невключ.:

Цикл для j от 0 ~~вкл~~ включ. до (N-2) невключ.:

sum := 0

цикл для i1 от i включ. до (i+2) включ.:

цикл для j1 от j включ. до (j+2) включ.:

sum := sum + r[i1][j]

конец циклов для i1 и j1

r[i+1][j+1] := sum/9 # средн. арифметическ.; не целочисленное деление

Конец цикла для j

Конец цикла для i

Конец

После окончания выполн. программы в перемен. r будет лежать искомый массив.

4) Не считая знакового разряда, в числах 1-й последовательности 7 разрядов, а 2-й – 15. Значит, в произведении таких чисел будет максимум (7+15)=22 двоичных разряда (а со знаковым разрядом – 23)

Количество произведений чисел: $8 \cdot 16 = 2^7$

Значит, сумма этих произведений в худшем случае займет ~~(22+7+1)~~ разрядов ~~или~~ 22+7+1=30 битов (т.к. 22-знач. число умножить на 7-знач. число в двоичн. системе даст максимум

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

29-знач. число; +1 из-за знакового разряда).
Значит, для каждой пары последовательностей результатом является одно 30-разрядн. число.
Для 1024 пар результатом являются 1024 30-разрядных числа.
Ответ: 1024; 30.

+10

Олимпиада школьников «Надежда энергетики»

№ группы	Место проведения
I 11F01	МЭИ с использованием ВКС

RO51-69

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ Плешивцев

ИМЯ Кирилл

ОТЧЕСТВО Михайлович

Дата рождения 21.06.2004 Класс: 11

Предмет ИНФОРМАТИКА Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 5 листах Дата выполнения работы: 20.02.2022
(число, месяц, год)

Подпись участника олимпиады:

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-69

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

1. Создадим пустую таблицу M на N, заполним её нулями. Введём переменную i и присвоим ей значение 1. Переменные x и y - номера строки и столбца текущей ячейки соответственно. Изначально x=y=1. В массиве steps сохраним четыре пары чисел: {0;1}, {1;0}, {0;-1}, {-1;0}. Переменная j будет изменяться от 0 до 3, принимая целые значения. Пара чисел steps[j] задаёт текущее направление движения по таблице. Пока i≤M·N будем выполнять следующие операции:

1) Присвоим table[x][y] значение i, table - таблица

2) Найдём координаты следующей ячейки:

x_next = x+steps[j], y-next = y+steps[j]

3) Если ячейка с координатами (x_next, y-next) существует и ещё не заполнена, т.е. 1≤x_next≤M and 1≤y-next≤N

~~and~~

and table[x_next][y_next] == 0

то x=x_next y=y_next

4) В противном случае необходимо сменить направление движения: j = (j+1)%4. Сложение осуществляется по модулю 4, поскольку j должна несколько раз пройти цикл от 0 до 3, чтобы обеспечить корректное заполнение таблицы. Координаты новой ячейки: x+= steps[j], y+= steps[j].

5) Увеличим i на 1, поскольку требуется заполнить таблицу последовательными натуральными числами.

Алгоритм завершит свою работу, когда последнее вписанное в таблицу значение будет равно M·N, т.е. будут заполнены все M·N ячеек таблицы.

Алгоритм изменит каждую ячейку ровно 1 раз (не считая инициализации таблицы), поэтому асимптотика O(M·N)

+10

2. Назовём массив точек a. Переберём все возможные тройки точек и найдём периметры соответствующих треугольников. Из найденных периметров выберем наибольший. Обозначим его res, а индексы соответствующих точек будем хранить в массиве dots.

1) В цикле переберём все $i \in [0; N)$ (предположим, что индексация ведётся с 0)
2) Во вложенном цикле переберём $j \in (i; N)$
3) В третьем вложенном цикле переберём $k \in (j; N)$
4) Найдём длины отрезков между точками:

```
ln1 = sqrt((a[i][0] - a[j][0])*(a[i][0] - a[j][0]) +
    +(a[i][1] - a[j][1])*(a[i][1] - a[j][1]) +
    +(a[i][2] - a[j][2])*(a[i][2] - a[j][2]))
ln2 = sqrt((a[i][0] - a[k][0])*(a[i][0] - a[k][0]) +
    +(a[i][1] - a[k][1])*(a[i][1] - a[k][1]) +
    +(a[i][2] - a[k][2])*(a[i][2] - a[k][2]))
ln3 = sqrt((a[j][0] - a[k][0])*(a[j][0] - a[k][0]) +
    +(a[j][1] - a[k][1])*(a[j][1] - a[k][1]) +
    +(a[j][2] - a[k][2])*(a[j][2] - a[k][2]))
```

sqrt(n) - функция взятия квадратного корня из n.

5) Если сумма ln1, ln2 и ln3 больше res, и при этом каждое из них меньше суммы двух других (треугольник невырожденный), обновим ответ

```
res = ln1 + ln2 + ln3
dots[0] = i
dots[1] = j
dots[2] = k
```

+10

Перед циклами проинициализируем res, dots[0], dots[1] и dots[2] нулями. Тогда, если по завершении работы алгоритма res==0, то все точки лежат на одной прямой. Иначе в dots будут находиться точки, задающие треугольник с наибольшим ~~периметром~~ периметром. Исходя из условия, достаточно найти один треугольник с наибольшим периметром, поэтому в dots и будет находиться ответ.

Алгоритм совершит не более чем N операций в каждом из трёх вложенных циклов, поэтому асимптотика $O(N^3)$

3. Пусть исходная матрица N×N называется a. Создадим пустую матрицу N×N. res

Двумя вложенными циклами от 0 до N-1 (предположим, что индексация с нуля) переберём все ячейки исходной таблицы. Посчитаем для ячейки с координатами (i, j) ~~сумму её соседей~~. ячейки с координатами (i, j), посчитав сумму её соседей и её ~~[illegible]~~ усреднённое значение, $\frac{X}{S}$ разделив на их количество (сама ячейка тоже учитывается).

- если ~~i==j==0~~ i==j==0, то $s = \frac{a[i][j] + a[i+1][j] + a[i][j+1] + a[i+1][j+1]}{4}$
- если i==0 и j==N-1, то $s = \frac{a[i][j] + a[i+1][j] + a[i][j-1] + a[i+1][j-1]}{4}$
- если i==N-1 и j==0, то $s = \frac{a[i][j] + a[i-1][j] + a[i][j+1] + a[i-1][j+1]}{4}$
- если i==j==N-1, то $s = \frac{a[i][j] + a[i-1][j] + a[i][j-1] + a[i-1][j-1]}{4}$
- если i==0 и 0<j<N-1, то

$$s = \frac{a[i][j] + a[i][j-1] + a[i][j+1] + a[i+1][j-1] + a[i+1][j] + a[i+1][j+1]}{6}$$

~~[illegible]~~

• если $i==N-1$ и $0<j<N-1$, то

$$s=\frac{a[i][j]+a[i][j-1]+a[i][j+1]+a[i-1][j-1]+a[i-1][j]+a[i-1][j+1]}{6}$$

• если $0<i<N-1$ и $j==0$, то

$$s=\frac{a[i][j]+a[i-1][j]+a[i+1][j]+a[i-1][j+1]+a[i][j+1]+a[i+1][j+1]}{6}$$

• если $0<i<N-1$ и $j==N-1$, то

$$s=\frac{a[i][j]+a[i-1][j]+a[i+1][j]+a[i-1][j-1]+a[i][j-1]+a[i+1][j-1]}{6}$$

• если $0<i<N-1$ и $0<j<N-1$, то

$$s=\frac{a[i][j]+a[i-1][j]+a[i+1][j]+a[i-1][j-1]+a[i][j-1]+a[i+1][j-1]+a[i-1][j+1]+a[i][j+1]+a[i+1][j+1]}{9}$$

Запишем s в ячейку res[i][j]. После выполнения алгоритма res будет являться ответом. Алгоритм обрабатывает каждую ячейку 1 раз, поэтому асимптотика $O(n^2)$.

+9

можно сгруппировать короче!

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-69

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

4. Рассмотрим произведение двух элементов. Один из них имеет 15 значащих +1 знаковый, а второй – 7+1.
Тогда их произведение имеет 23 разряда, включая знаковый*
Всего попарных произведения элементов последовательностей из 16 и 8 элементов
$8 \cdot 16 = 2^3 \cdot 2^4 = 2^7$
Пар последовательностей 1024 => всего необходимо хранить $2^7 \cdot 1024 = 2^7 \cdot 2^{10} = 2^{17}$ чисел

Ответ: 2^{17} чисел разрядности 23

* Число с 16-ю разрядами по модулю не больше $2^{15}-1 = 32768 - 1 = 32767$

Число с 8-ю разрядами по модулю не больше $2^7 - 1 = 127$

$32767 \cdot 127 = 4161409$

$4161409_{10} = \underbrace{1111110111111111000000}_{22\ разряда}1_2$

22 разряда + 1 знаковый разряд = 23 разряда

(7 + 15) разрядов → 2^7 суммируется → 29 разрядов + 1 знак = 30 р

+7

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-69

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

5. $\frac{\mathcal{E}}{R+r} = I$, I - сила тока
$\mathcal{E}$ - ЭДС
R - сопротивление нагрузки
r - ~~си~~ внутреннее сопротивление источника ЭДС

$\mathcal{E} = I(R+r) = IR + Ir = U + Ir$ $\begin{cases} \mathcal{E} = U_1 + I_1 r \\ \mathcal{E} = U_2 + I_2 r \end{cases} \Rightarrow r = \frac{U_2 - U_1}{I_1 - I_2}$

$\mathcal{E} = U + Ir$

Пусть a - матрица из $2N$ строк и 2 столбцов: напряжение и сила тока в i-ом эксперименте соответственно. Заведём динамические массивы du и di. Будем хранить в них модули попарных разностей: для $|U_i - U_j|$ в du и $|I_i - I_j|$ в di соответственно для $\forall \{i, j\} \subset [0; 2N)$, $i \neq j$. Будем предполагать, что входные данные корректны и что. С помощью алгоритма Евклида найдём НОК всех элементов di. Обозначим его за t. Мысленно приведём все дроби ~~к общему~~ вида $\frac{du_i}{di_i}$ к общему знаменателю t. ~~будем~~ и сложим. Сумма числителей $S = \sum_{i=0}^{2N-1} du_i * (di_i \,//\, t)$, ~~//~~ - целочисленное деление.

$S/(2N*t)$ - среднее τ. Заметим, что вещественное деление (1) появилось только ~~на~~ на данном этапе алгоритма, до этого все вычисления оставались целочисленными. Таким образом, точность вычисления намного выше, чем если бы τ считались для каждой пары экспериментов отдельно.
Подставим найденное τ в каждый эксперимент ($\varepsilon = u + I * \tau$) и найдём сумму всех ε_i. Разделим на $2N$ и получим среднее ε.

+10

Олимпиада школьников «Надежда энергетики»

I11F01	Дистанционно, с использованием ВКС
№ группы	Место проведения

RO51-73

шифр

← Не заполнять. Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ Птицын

ИМЯ Илья

ОТЧЕСТВО Константинович

Дата рождения 04.04.2005 **Класс:** 11

Предмет Информатика **Этап:** Заключительный

Работа выполнена на 5 **листах** **Дата выполнения работы:** 20.02.2022 (число, месяц, год)

Подпись участника олимпиады: Пт

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ ⇨ RO51-73

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

Задача 1

+10

A - массив (изначально заполненный нулями)

f=0

i=1

j=1

t=1

Пока t ≤ M·N: A[i][j]=t ; t=t+1

если (f=0 и A[i+1][j] ≠ 0) или (f=1 и A[i][j+1] ≠ 0) или (f=2 и A[i-1][j] ≠ 0) или (f=3 и A[i][j-1] ≠ 0):

f=(f+1) mod 4

если f=0:

i=i+1

если f=1:

j=j+1

если f=2:

i=i-1

если f=3:

j = j - 1

Задание 4

16 разрядов - число < 32768

8 разрядов - число < 128

Для каждой пары последовательностей результат < ~~[illegible]~~

32768 · 128

необходимо ~~[illegible]~~ 1024 30-разрядных чисел

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ ⇨ RO51-73

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

Задание 2

$i = 0$

A - исходный массив

$m = 0$

Пока $i < N$:

  $j = i$

  Пока $j < N$:

    $k = j$

    Пока $k < N$:

      $L_1 = \sqrt{(A[i][0] - A[j][0])^2 + (A[i][1] - A[j][1])^2 + (A[i][2] - A[j][2])^2}$

      $L_2 = \sqrt{(A[j][0] - A[k][0])^2 + (A[j][1] - A[k][1])^2 + (A[j][2] - A[k][2])^2}$

      $L_3 = \sqrt{(A[i][0] - A[k][0])^2 + (A[i][1] - A[k][1])^2 + (A[i][2] - A[k][2])^2}$

      если $L_1 \geq L_2$ и $L_1 \geq L_3$ и $L_2 + L_3 \leq L_1$:

        продолжить цикл $k = k+1$

      если $L_2 \geq L_1$ и $L_2 \geq L_3$ и $L_1 + L_3 \leq L_2$:

        $k = k+1$

        продолжить цикл

      если $L_3 \geq L_1$ и $L_3 \geq L_2$ и $L_1 + L_2 \leq L_3$:

        $k = k+1$

        продолжить цикл

      если $L_1 + L_2 + L_3 > m$:

        $m = L_1 + L_2 + L_3$

        $x_1 = i$

        $x_2 = j$

        $x_3 = k$

      $k = k+1$

    $j = j+1$

  $i = i+1$

Треугольник $x_1; x_2; x_3$ имеет самый большой периметр m

x10

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ ⇨ RO51-73

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

A - исходный массив (Задание 3)

```
i=0
Пока i < N:
  j=0
  Пока j < N:
      s=0
      k=0
      если i < N-1:
          s = s + A[i+1][j]
          k = k+1
      если i > 0:
          s = s + A[i-1][j]
          k = k+1
      если j > 0:
          s = s + A[i][j-1] ; k = k+1
      если j < N-1:
          s = s + A[i][j+1]
          k = k+1
      если i > 0 и j > 0:
          s = s + A[i-1][j-1]
          k = k+1
      если i > 0 и j < N-1:
          s = s + A[i-1][j+1]
          k = k+1
      если i < N-1 и j > 0:
          s = s + A[i+1][j-1]
          k = k+1
      если i < N-1 и j < N-1:
          s = s + A[i+1][j+1]
          k = k+1
      s = s + A[i][j]
      k = k+1
      A[i][j] = s : k
      j = j+1
  i = i+1
```

+10

Олимпиада школьников «Надежда энергетики»

Вариант: 37 111

ШИФР, НЕ ЗАПОЛНЯТЬ ⇨ RO51-73

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

Задание 4

16 разрядов – число < 32768

8 разрядов – число < 128

для каждой пары последовательностей результат < 32768 · 128 · 128

необходимо 1024 30-разрядных числа

+10

Олимпиада школьников «Надежда энергетики»

Вариант: 37111 ШИФР, НЕ ЗАПОЛНЯТЬ ⇒ RO51-73

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

Задание 5

I_1, U_1 - первый эксперимент

I_2, U_2 - второй эксперимент

$R_1 = U_1 : I_1$

$R_2 = U_2 : I_2$

$$r = \frac{I_1 \cdot R_1 - I_2 \cdot R_2}{I_2 - I_1}$$

$$\varepsilon = I_1 \cdot (R_1 \cdot r)$$

r – внутреннее сопротивление

ε – ЭДС

надо учесть большее число опытов и усреднение результатов

+8

Олимпиада школьников «Надежда энергетики»

№ группы	Место проведения
I9F-01	Дистанционно, с использованием ВКС

WZ73-94

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 37991

ФАМИЛИЯ Семенова

ИМЯ Софья

ОТЧЕСТВО Андреевна

Дата рождения 24.12.2006

Класс: 9

Предмет Информатика

Этап: Заключительный

Работа выполнена на 6 листах

Дата выполнения работы: 20.02.2022
(число, месяц, год)

Подпись участника олимпиады: Семенова

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ WZ73-94

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

Задача 1

$5E_n$

3 разряда в 10СС

оканчивается на 0 в 8СС

n_{min} — ?

Заметим для начала, что $n \geq 15$, т.к. E в 10-чной системе счисления — 14.

Переведем 5E в 10-чную систему счисления:

$$5E_n = E \cdot n^0 + 5 \cdot n^1 = (14 + 5n)_{10}$$

По условию, $14 + 5n \geq 100$

$5n \geq 86$

$n \geq 17{,}2$ (т.к. n — натуральное число, то минимально возможное n — 18)

Т.к. $5E_n$ в 8-чной системе счисления оканчивается на 0, значит $(14+5n)_{10}$ делится нацело на 8. Значит (у числа 14 остаток от деления на 8 = 6) остаток от деления 5n на 8 должен быть = 2.

Переберем различные варианты $n \geq 18$, и получим, что при $n = 18$ все условия выполняются.

$5E_{18} = 5 \cdot 18 + 14 = 104_{10}$ (= 3 разряда)

$104_{10} = 150_8$ (оканчивается на 0)

$$\begin{array}{l} 104 \,|\, 8 \\ \text{ост. } 0 \quad 13 \,|\, 8 \\ \quad\quad \text{ост. } 5 \quad 1 \,|\, 8 \\ \quad\quad\quad\quad \text{ост. } 1 \quad 0 \end{array}$$

+10

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

Задача 2

Так как необходимо найти наименьшее количество кусков, а не наименьшую суммарную их длину, то можно спокойно выбирать куски в порядке убывания длины (так как чем длиннее будет каждый провод, тем меньше проводов понадобится)

Примерный код алгоритма. (python)

```
M = int(input())
N = int(input())
arr = []  #массив длин кусков
K=0  #счетчик минимального количества кусков
j=N-1  #итератор цикла while (необходим для подсчета длин от наибольшей к наименьшей)
cur_len=0 #счетчик текущей длины
for i in range(N):
    arr.append(int(input())
arr.sort()
while cur_len < M:
    cur_len += arr[j]
    j -= 1
    K+=1
print(K)
```

X 10

То есть надо отсортировать длины всех кусков и пройтись от наибольшей к наименьшей. Проходить надо до того момента, пока суммарная длина всех ~~пр~~ пройденных кусков (включая текущий) не станет ≥ M. Количество кусков, по которым мы прошлись и будет минимальным.

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

Задача 3

В задаче требуется найти среднее значение элемента в массиве, затем найти разницу ~~к~~ каждого элемента со средним значением и посчитать среднее значение всех разниц (их тоже будет N, как и элементов). А затем пройтись по значениям разниц и посмотреть, что если эта разница в 2 раза отличается от среднего значения разниц, то мы нашли один из требуемых результатов.

+10

Примерный код (python).

```
#введем переменные:
# N- длина массива вводимых чисел и массива отклонений
arr=[] # массив вводимых чисел
otkl=[] # массив отклонений
res=[] # массив, куда будем добавлять выделенные в соответствии с условием результаты
summ=0 # для подсчета суммы массива вводимых чисел
summ_otkl=0 #для подсчета суммы массива отклонений
#Mid - среднее значения элемента в массиве вводимых чисел
#Mid_otkl - среднее значения элемента в массиве отклонений


#сама программа:
N=int(input())
for i in range(N):
  arr.append(int(input()))

for i in range(N):
  summ +=arr[i]      # поиск среднего значения элемента в массиве вводимых чисел
Mid =summ/N

for i in range(N):
    otkl.append(abs(arr[i]-Mid))    # поиск среднего значения разниц каждого вводимого числа со средним значением массива вводимых чисел
    summ_otkl +=abs(arr[i]-Mid)
Mid_otkl = summ_otkl/N

for i in range(N):
   if (otkl[i] > Mid_otkl*2):
      res.append(arr[i])
print(*res)
```

Задача 5

По закону Ома $I = \frac{U}{R}$, т.е. I и U связаны линейной зависимостью. Значит при постоянном сопротивлении R в случае уменьшения I уменьшается и U, т.е. $I_1 < I_2$

$U_1 < U_2$ (*)

Массив токов будет правильным, когда элементы массивов тока и напряжения с одинаковыми индексами будут соответствовать одному измерению. Учитывая утверждение (*), для этого достаточно отсортировать массив напряжений по убыванию

+10

Примерный код сортировки массива напряжений по убыванию (python):

```
n = len(U) # U - массив напряжений
unordered = True
while unordered:
    unordered = False
    for i in range(n-1):
        if U[i] < U[i+1]:
            U[i], U[i+1] = U[i+1], U[i]
            unordered = True
    n -= 1
```

Олимпиада школьников «Надежда энергетики»

Вариант: 37991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ WZ73-94

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

Задача 4

1. Особенностью операции умножения n-разрядных двоичных сомножителей является увеличения разрядности произведения до $n+n=2n$.
При умножении двух 16-разрядных чисел могут получаться ~~[illegible]~~ двоичные числа до 32-х разрядов. То есть умножение восьми 16-разрядных чисел равносильно умножению четырех 32-х разрядных, то есть двух 64-разрядных ~~[illegible]~~ Получаем результат до 128 разрядов, то есть 2^7 разрядов.
Аналогично, при умножении восьми 8-разрядных двоичных чисел могут получаться двоичные числа до 2^6 разрядов.

2. Особенностью сложения двух двоичных чисел, большее из которых имеет n разрядов, является увеличения разрядности до $n+1$. => при сложении чисел разрядностью 2^6 и 2^7 могут получаться двоичные числа до 2^8 разрядов.

×10

3) Из пунктов 1 и 2 следует, что для одной пары последовательностей необходимо хранить 2 результата:

1. текущее произведение разрядности 2^7
2. сумму произведений разрядности 2^8

Сначала в ячейках памяти, отведенных под текущее произведение, считаем произведение чисел одной последовательности (оно не превысит 2^7 разрядов, поэтому именно 2^7 разрядов под него надо выделить). Затем это произведение отправляем в ячейки памяти, отведенные под сумму. Далее нам хранить его отдельно не надо, поэтому считаем произведение второй последовательности в тех же ячейках памяти, которые отведены для подсчета текущего произведения (разрядность тоже не превысит 2^7). Потом второе произведение прибавляем к первому, которое уже хранится в ячейках памяти, отведенных под сумму произведений последовательностей.

Продолжение на следующем листе.

Олимпиада школьников «Надежда энергетики»

Вариант: 37991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ WZ73-94

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

Так как на вход подаются 1024 пары последовательностей, то всего получится 1024 суммы произведений. Значит нужно однозначно хранить как минимум 1024 результата разрядностью 2^8. В пункте 3 доказано, что необходим еще один результат разрядностью 2^7 для подсчета произведений (для одной пары последовательностей). Для 1024 пар тоже понадобится только один результат разрядностью 2^7 для подсчета произведений: сначала поочереди считаем произведения в одной паре последовательностей (описано в пункте 3), затем аналогично делаем со второй парой последовательностей, с третьей, и так далее, до последней пары.

Итого, нам понадобится 1025 результатов. 1 число разрядностью 2^7 для ~~кроме~~ подсчета произведений поочереди, и 1024 числа разрядностью 2^8 для хранения и подсчета каждой из сумм.

Олимпиада школьников «Надежда энергетики»

I10F01	Дистанционно, с использованием ВКС
№ группы	Место проведения

QZ94-78

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 37101

ФАМИЛИЯ СОКОЛОВ

ИМЯ АРТЁМ

ОТЧЕСТВО МИХАЙЛОВИЧ

Дата рождения 18.01.2006

Класс: 10

Предмет ИНФОРМАТИКА

Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 4 листах

Дата выполнения работы: 20.02.2022
(число, месяц, год)

Подпись участника олимпиады:

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37101 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QZ94-78

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

№1

число $K = 206 \Rightarrow n \geq 7$

K записано в n системе счисления

в восьмеричной с.сч. заканчивается на $0 \Leftrightarrow K \vdots 8$

в тринадцатеричной с.сч. заканч. на $0 \Leftrightarrow K \vdots 13$

в десятичной с.сч. имеет 3 разряда (т.е < 1000 и > 99)

Определим верхнюю границу для n.

Предположим, что $n \leq 23$.

Тогда при $n = 23$ ~~K десят~~ $K_{10} = 2 \cdot 23^2 + 6 \cdot 23^0 = 2 \cdot 529 + 6 \cdot 1 = 1064$ – противоречие (4 разр.)

Если $n \leq 22$:

Тогда при $n = 22$: $K_{10} = 2 \cdot 22^2 + 6 \cdot 22^0 = 2 \cdot 484 + 6 \cdot 1 = 974$

Значит $n \leq 22$. Переберём n „сверху вниз“, ~~заканчивая~~ пока не найдём, такое n, что $K_{10} \vdots 8$ и $\vdots 13$.

$(K_{10} = 2 \cdot n^2 + 6 \cdot n^0)$

при $n = 22$:

$K_{10} = 974 \not\vdots 8 \Rightarrow$ не подх.

при $n = 21$:

$K_{10} = 887 \not\vdots 8 \Rightarrow$ не подх.

при $n = 20$:

$K_{10} = 806 \not\vdots 8 \Rightarrow$ не подх.

при $n = 19$:

$K_{10} = 728 \vdots 8$ и $\vdots 13 \Rightarrow$ подходит.

Тогда наибольшее $n = 19$.

т.к. K_{10} при $n = 19$ нацело делится на 13 и 8 и имеет три разряда

Ответ: 19

+10

№2

Алгоритм:

1. Вводится M, N.
2. Вводится последовательность из N чисел, записывается в массив S.
3. Сортируется массив S по возрастанию.
4. Заведём переменную ~~m~~ count = 0 и переменную mm=0
5. Идём циклом по массиву S:

    mm+= i (где i – элемент массива в итерации)

    count += 1 ← одна итерация

    Если mm ≥ M:

        выводим count

        завершаем программу

конец алгоритма

+10

~~С~~ Т.к. в условии нет особых пометок, считаю что при производстве разрешено резать провода.

№3

Алгоритм:

1. Вводится N
2. Вводится массив S из N чисел.
3. ~~Заведём числовую переменную K~~
4. Скопируем массив S в массив m.
5. Идём циклом по массиву S (i – индекс) (индексация с 1

    ~~Если~~ Если i==1:

        ~~K = S[i]~~

        $S[i] = \frac{m[i+1]+m[i+2]+m[i+3]}{3}$

    Если i==N:

        $S[i] = \frac{m[i-1]+m[i-2]+m[i-3]}{3}$

    Если i==N-1:

        $S[i] = \frac{m[i+1]+m[i-1]+m[i-2]}{3}$

    Если N-1 > i > 1:

        $S[i] = \frac{m[i+1]+m[i-1]+m[i+2]}{3}$

6. Вывод массива S.

конец алгоритма

+10

Олимпиада школьников «Надежда энергетики»

Вариант: 37101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QZ94-78

Лист [ ] из [ ]

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

№4

Две последовательности:

b - 8 чисел по 16 дв. разр.

d - 16 чисел по 8 дв. разр.

Всего 1024 пары последовательностей

Оценим сверху число $K \in b$:

не более 1111111111111111 в двоичной записи, то есть не более 65535

Оценим сверху число $m \in d$:

не более 11111111 в двоичной записи, то есть не более 255

128 сложений
$\times 10$

Тогда т.к не более $255 \cdot 65535 =$
$255 = 2^8 - 1, 65535 = 2^{16} - 1$ $= 16711425$
$\Rightarrow 16711425$ имеет 24 разр. в дв. зап.
При попарном произведении всего
$8 \cdot 16$ произведений то есть итоговая
сумма попарных произведений не больше
имеет 24 разр. в дв. записи.
$8 \cdot 16 \cdot 16711425 = 2^7 \cdot 16711425$

добавит 7 разрядов в дв. записи
Тогда сумма имеет не более $24 + 7 = 31$ разряда в двоичной
записи.
От каждой пары последовательностей результат -
1 число $\Rightarrow$ число результатов = числу пар посл.
Тогда всего 1024 результата по 31 разряду
Ответ: 1024 числа, 31 разряд в дв. записи.

Олимпиада школьников «Надежда энергетики»

Вариант: 37101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QZ94-78

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

№5

$I=\frac{U}{R}$ $\mathcal{E}=12В$

$\mathcal{E}=\frac{U_2I_1-U_1I_2}{I_1-I_2}$

Алгоритм: 1. Ввод I_1, I_2, U_1, U_2

2. Заведём функцию get_e(U_1, U_2, I_1, I_2):

return $\frac{U_2I_1-U_1I_2}{I_1-I_2}$

3. Запишем в переменную min минимальное число от выполнения функции get_e, а в max максимальное число от выполнения функции get_e в 16 случаях, когда каждая переменная домножается на $\frac{100}{99}$ или $\frac{100}{101}$ (на 4 переменных 16 вариантов).

4. Если min >12 или max <12:

вывести "измерения проведены неверно"

Иначе:

вывести "измерения проведены верно"

Конец алгоритма

Откуда? Как не получится? А как не 1%?

+6

Олимпиада школьников «Надежда энергетики»

№ группы	Место проведения
I9F01	Дистанционно с проведением ВКС

WZ73-32

шифр

← Не заполнять. Заполняется ответственным работником

Вариант № 37991

ФАМИЛИЯ Храновская

ИМЯ Марина

ОТЧЕСТВО Евгеньевна

Дата рождения 04.02.2006 **Класс:** 9

Предмет информатика **Этап:** заключительный

Работа выполнена на 2 **листах** **Дата выполнения работы:** 22.02.2022
(число, месяц, год)

Подпись участника олимпиады: МХ

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

SE # 514 " "

Олимпиада школьников «Надежда энергетики»

Вариант: 37991

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇒ WZ73-32

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

№1

(дано n СС)

Число 514_n в десятичной системе счисления имеет 3 разряда, т.е. оно ≥100, в восьмой СС заканчивается 0, т.е. кратно 8.

~~Наверняка~~ n не может быть 5, т.к. 5 и меньше, т.к. ~~...~~ число в такой-либо СС ~~...~~ содержит цифры, которые не больше этой СС ⇒ n > 5

Пусть n = 6, тогда должно быть кратно 8. $514_6 = 276_8$ — число не кратно 8 (последняя цифра не 0), противоречие, т.е. значит n ≠ 6

Пусть n = 7, $514_7 = 400_8$ — число кратно 8

$514_7 = 256_{10}$ — число имеет 3 разряда (≥100)

Все условия выполняются ⇒ n = 7

Ответ: n = 7

№3

Есть последовательность чисел длиной N: $a = \{a_1, a_2, a_3, \dots, a_N\}$

Шаг 1. Вычислим среднее значение A

$A = \frac{\sum a_i}{N}$

Шаг 2. Заведем массив B, в нем будут храниться отклонения от A

Для каждого i от 1 до N: $b_i = |a_i - A|$

Шаг 3. Вычислим среднее значение отклонения B

$B = \frac{\sum b_i}{N}$

Шаг 4. Выделим нужные результаты

Для каждого i от 1 до N

Если $\frac{b_i}{B} > 2$ или $\frac{b_i}{B} < \frac{1}{2}$, то b_i добавляем в ответ

Иначе b_i не является нужным результатом

+B

не ...

+9

а если B = 0?

Лист 2 из

Олимпиада школьников «Надежда энергетики»

Вариант: 37991

ШИФР, НЕ ЗАПОЛНЯТЬ! WZ73-32

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

№4

При перемножении первого и второго чисел результат может занимать максимум 24 разряда (в случае когда в двух числах все разряды единицы). То есть для каждой пары надо хранить 3 24разрядных числа. Сумма этих произведений может занимать 27 разрядов. Для каждой пары надо хранить 1 27разрядное число

Пар 1024, т.е. всего надо хранить 1024 · 24 = 24576 24разрядных числа и 1024 27разрядных числа

№5

N — кол-во измерений

$R = \frac{U}{I}$ Пусть I — массив значений тока $I = \{I_1, I_2, \ldots, I_n\}$

Пусть U — массив значений напряжения $U = \{U_1, U_2, \ldots, U_n\}$

Сопоставим каждому элементу массива U его индекс в массиве

$U = \{(U_1, 1), (U_2, 2), \ldots, (U_n, n)\}$ $ans = \{0, 0, \ldots, 0\}$ — будущий массив токов (пока еще пустой)

Отсортируем массив U по ~~[illegible]~~ убыванию значений

Для каждого i от 1 до N:

$ans_{U_i.инд} = I_i$

Таким образом, для каждого U_i будет I_i.

Массивы отсортированы в таком порядке, т.к. при увеличении напряжения ток тоже должен увеличиваться, чтобы сопротивление оставалось неизменным

№2 I способ

Динамическое программирование (Динамика)

Динамическое программирование с количеством кусков

Пусть dp — массив с ~~возможными длинами~~ количеством кусков, индексы которого — возможные длины $dp = \{\ldots, \infty\}$

Пусть a — массив с длинами кусков $a = \{a_1, a_2, \ldots, a_n\}$ размера N

M — искомая длина

Для каждого j от 1 до N:

$dp[j] = 1$

Для каждого i от 1 до M:

Для каждого j от 1 до N:

$dp[i + a_j] = \min(dp[i + a_j], dp[i] + 1)$

ответ лежит в $dp[M]$

II способ. Бэктрекинг. Перебор всех вариантов. Представим массив a как двоичное число, где 1 — если мы берем этот кусок (учитываем его), а 0 — не берем. Перебираем 2^N таких вариантов, и считая каждый раз ответ для текущего подмножества (сумму и кол-во кусков). Выбираем ответ с длиной равной M и минимальным кол-вом кусков.

10

×10

правильно

не правильно

Лист 02 из 02

Олимпиада школьников «Надежда энергетики»

№ группы	Место проведения
ЗИФО1	Дистанционно, с использованием ВКС

Шифр: RO51-16

← Не заполнять. Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ ХРУСТАЛЕВ

ИМЯ Влад

ОТЧЕСТВО Николаевич

Дата рождения 21.09.2004 **Класс:** 11

Предмет Информатика **Этап:** Заключительный

Работа выполнена на 5 **листах** **Дата выполнения работы:** 20.02.2022 (число, месяц, год)

Подпись участника олимпиады: Хруст

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

Олимпиада школьников «Надежда энергетики»

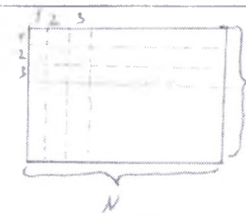
Вариант: 37111

ШИФР. НЕ ЗАПОЛНЯТЬ!

RO51-16

№ 1

Пусть таблица имеет такой вид



M, а его координаты [X, Y]

X – по оси с макс. зн. M

Y – по оси с макс. зн. N

Изначально пусть таблица имеет значения „-1" в каждой клетке

Далее алгоритм

F – переменная описывающая направление движения

Ввод X = 1; Y = 1; F = 1

Ввести значение в [X, Y]

+10?

```
for (i = 1; i < M N; i++).
  Если F == 1 // дв-е вправо
    Если y+1 <= N и [X, y+1] == -1
      y = y+1
      Вводим значение в [X; y]
    Если y+1 > N или [X; y+1] != -1
      F = 2
  Если F == 2 // дв-е вниз
    Если X+1 <= M и [X+1, y] == -1
      X = X+1
      Вводим значение в [X; y]
    Если X+1 > M или [X+1, y] != -1
      F = 3
  Если F == 3 // дв-е влево
    Если y-1 >= 1 и [X; y-1] == -1
      y = y-1
      Вводим значение в [X; y]
    Если y-1 < 1 или [X; y-1] != -1
      F = 4
  Если F == 4 // дв-е вверх
    Если X-1 >= 1 и [X-1, y] == -1
      X = X-1
      Вводим значение в [X; y]
```

Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-16

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

Если $x-1<1$ или $[x-1; y] != -1$:

F = 1.

Конец программы.

Таким образом таблица должна заполняться как нам нужно

№ 3.

Т.к. по краям матрицы 9-ти соседей не будет, то сделаю предположение, что берется ср. значение из имеющихся чисел.

Подматрица (3x3) будет иметь вид и коорд.

x-1 y-1	x-1 y	x-1 y+1
x y-1	X y	x
x+1 y-1	x+1 y	x+1 y+1

Изначальная матрица будет иметь название A, её координаты A[x; y];

Для вывода новых значений создадим матрицу B, с размерами N x N

Далее алгоритм:

```
for (x = 1, x ≤ N; x++):
    for (y = 1; y ≤ N ; y++):
        K=0 //считает кол. эл-ов в подматрице
        sum =0 // сумма этих эл-ов.
        for (i = x-1; i <= x+1, i++):
            for (j = y-1; j <= y+1; j++)
                if i >= 1 и j >= 1 и i ≤ N и j ≤ N:
                    k += 1
                    sum += A[i;j].

        B[x,y] = sum / K (деление)
```

× 10

Конец программы:

На выходе этого алгоритма получаем матрица $B(N \times N)$ с нужными значениями.

Олимпиада школьников «Надежда энергетики»

Вариант: 37 111

ШИФР, НЕ ЗАПОЛНЯТЬ! RO51-16

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

№4.

8 кр. {x, y, ..} × 16 8р {y, y, .., ..} => $8 \cdot 16 = 128$ чисел при попарных перемножениях с разрядностью максимальной 17, т.к. при сложении 15 разрядов (1_6) + 7р (1_3) = 16 разряд максимум + 1 разряд на знак

15+7 = 22 (17) + знак = 23

Далее складываются 128 чисел 17-ти разрядных, где 7-ой разряд - знак.

=> (128 чисел 16 разрядных сложение) +

128 чисел из 16-ти 1ц в двоичной записи сложение - самый максимум

=> $\frac{128}{2} + 16 = 64 + 16 = 80$ разрядов + 1 на знак => 81 разряд у числа равного сумме всех попарных пр-ий в 2-ух последовательностях

Если ~~все~~ на вход подается 1024 пара подряд и вычисление происходит по разр. у каждой пары, то есть 1n(1) — 1n(2), 2n(1) — 2n(2 => Ответ: 1024 числа 81 разрядных

30 р

Кольцо

Или же если вопрос имеет виду, сколько чисел должно храниться в памяти во время вычислений, то 128 чисел 17 разрядных; 1 число 81 разрядное для каждой пары

Ответ: 1024 чисел 81 разрядных

128 слагаемых дает +7 разрядов

23+7 = 30

+6

Лист 03 из 05

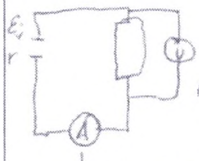
Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-16

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

№5.



Выведем ф-у $\mathcal{E}_i$ – ЭДС источника и r – внутреннее сопр. источника.

$$I = \frac{\mathcal{E}_i}{R+r}; \quad I = \frac{U}{R} \Rightarrow R = \frac{U}{I}$$

$$I = \frac{\mathcal{E}_i}{\frac{U}{I}+r} \Rightarrow I = \frac{\mathcal{E}_i I}{U+rI} \Rightarrow U+rI = \mathcal{E}_i$$

$$U + rI = \mathcal{E}_i$$

U – замеренное напряжение

I – измеренная сила тока, т.к. 2-е неизвестные, то чтобы найти их значения нужно использовать пару опытов

$$U_1 + rI_1 = \mathcal{E}_i = U_2 + rI_2 \qquad U + rI = \mathcal{E}_i \Rightarrow r = \frac{\mathcal{E}_i - U}{I}$$

$$r(I_1 - I_2) = U_2 - U_1 \qquad \frac{\mathcal{E}_i - U_1}{I_1} = \frac{\mathcal{E}_i - U_2}{I_2}$$

$$r = \frac{U_2 - U_1}{I_1 - I_2}$$

формула 1

$$\mathcal{E}_i I_2 - U_1 I_2 = \mathcal{E}_i I_1 - U_2 I_1$$

$$\mathcal{E}_i (I_2 - I_1) = U_1 I_2 - U_2 I_1$$

+10

$$\varepsilon_i = \frac{u_1 I_2 - u_2 I_1}{I_2 - I_1}$$

формула 2

Для более точного вычисления сделаем ~~ср~~ вычисление ср. знач. r и ε_i, по результатам $2N$ - экспериментов.

$(2N-1)\cdot(2N-2)\cdot(2N-3)\ldots(2N-(2N-1)) = a$

Всего значений будет $\cancel{(2N)\cdot(2N-1)}$, т.к. столько пар можно взять экспериментов и для каждого будет свое r и ε_i. После нахождения r и ε_i для каждой пары вычислим ср. знач. по формуле $\dfrac{\varepsilon_{i12} + \varepsilon_{i13} + \ldots}{\cancel{2N(2N-1)}\, a} = \varepsilon_{i\,ср.}$

и $\dfrac{r_{12} + r_{13} + \ldots + r_{(N-1)N}}{\cancel{2N(2N-1)}\, a} = r_{ср.}$

№2.

N строк по 3 столбца => N точек.

Расстояние м/у двумя точками равно $\sqrt{(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2}=R$

Всего возможно пр-ков: $C=\frac{N!}{3!(N-3)!}=\frac{N!}{6(N-3)!}$

Для $N=10$: $C_{10}=120$ => т.к. N большое число, то $C>>N$ ~~[illegible] перебор не выгоден~~

Обозначим формулу $R[i;j]=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2+(z_i-z_j)^2}$.

Далее алгоритм.

```
max = 0
for (i = 1; i ≤ (N-2); i++):
    for (j = i+1; j ≤ (N-1); j++):
        for (g = j+1; g ≤ N; g++):
            P = R[i;j] + R[i;g] + R[j;g].
            Если max < P:
                max = P;
```

вывести – max

Конец программы.

Нужное нам значение в конце программы – это max

Нет проверки существования Δ

+7

Олимпиада школьников «Надежда энергетики»

11Р01	Дистанционно с использованием ВКС
№ группы	Место проведения

RO51-27

шифр

Не заполнять
Заполняется ответственным работником

Вариант № 37111

ФАМИЛИЯ Чикунов

ИМЯ Данила

ОТЧЕСТВО Сергеевич

Дата рождения 23.04.2004 Класс: 11

Предмет Информатика Этап: Заключительный

Работа выполнена на 5 листах Дата выполнения работы: 20.02.2022
(число, месяц, год)

Подпись участника олимпиады: [подпись]

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

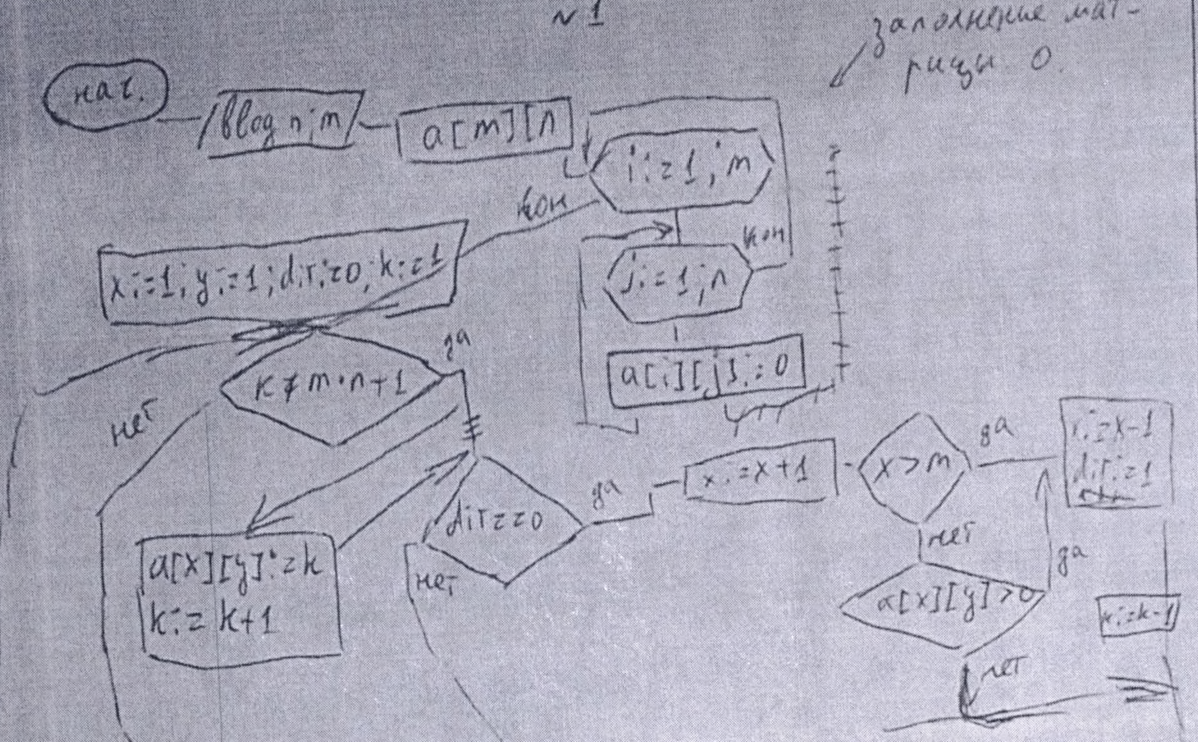
Олимпиада школьников «Надежда энергетики»

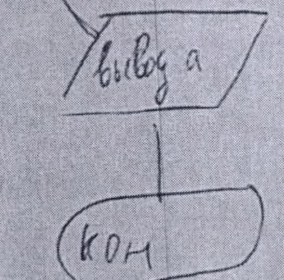
Вариант: 37111

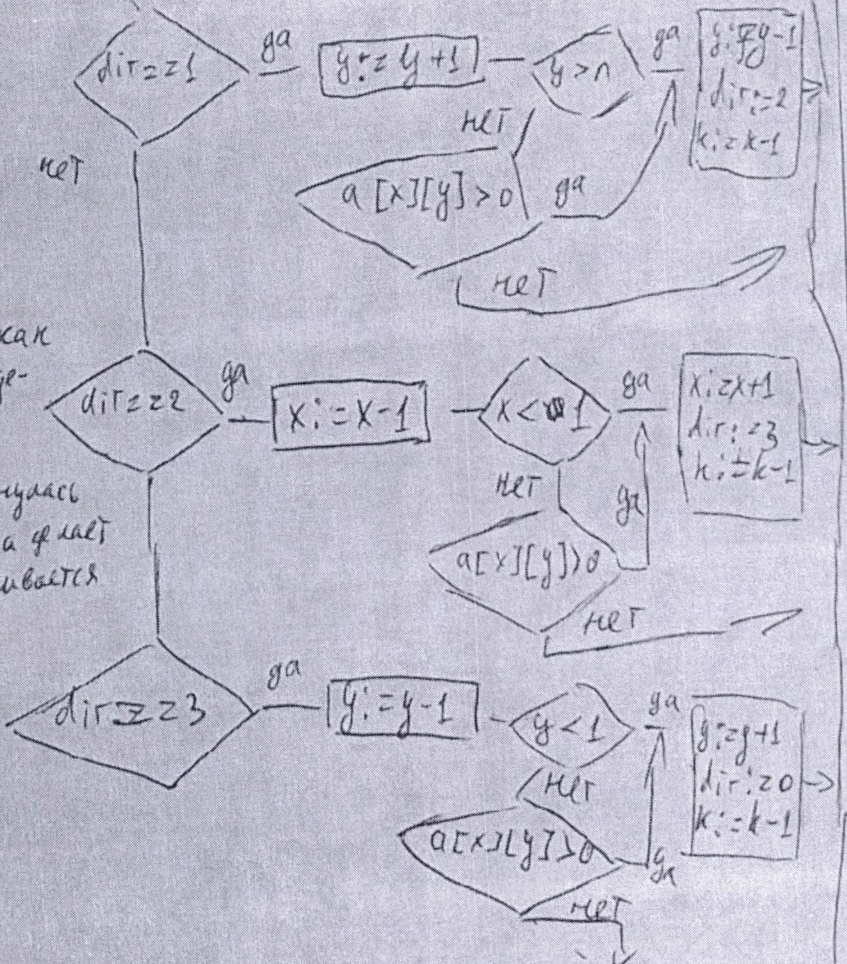
ШИФР, НЕ ЗАПОЛНЯТЬ! ⇒ RO51-27

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

№ 1







Алгоритм работает как змейка. Каждый ход делается шаг вперёд, а если змейка наткнулась на препятствие, то она делает шаг назад и поворачивается на 90° вправо.

+10

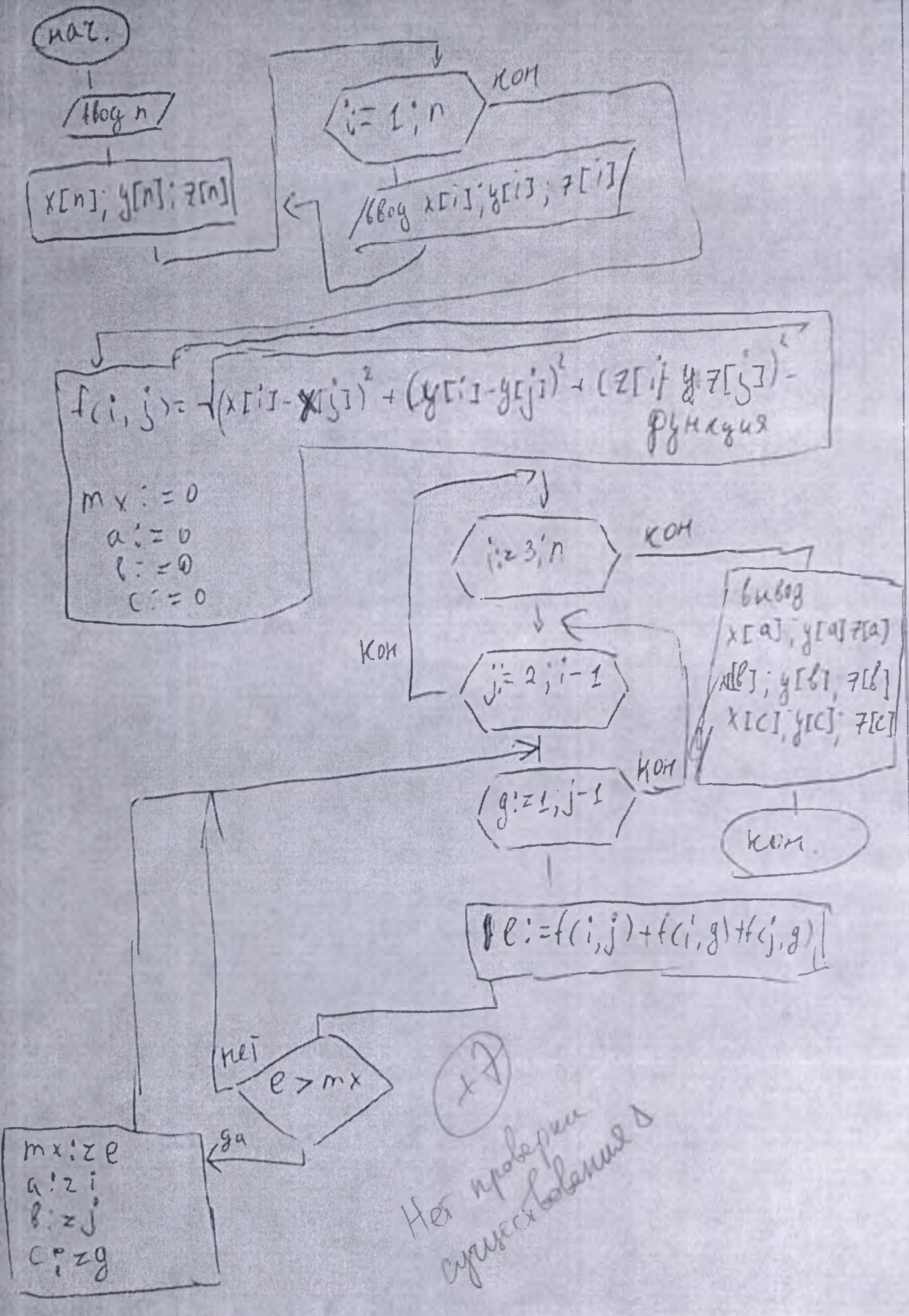
Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ RO51-27

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

№2



+7

Нет проверки существования Δ

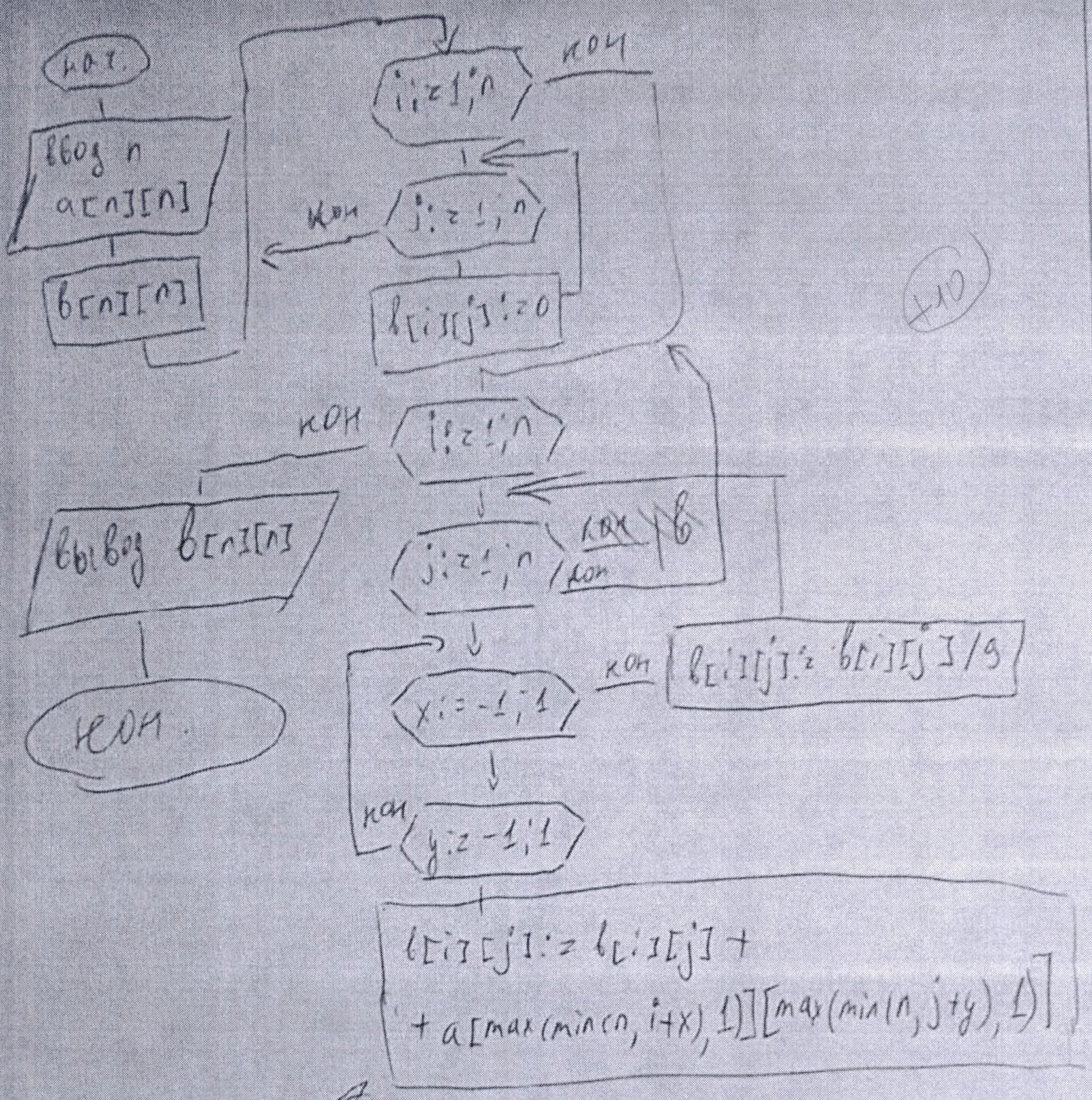
Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇒ RO51-27

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

№3



max – функция, возвращающая наибольшее из двух значений.

min – работает аналогично ~~max~~, но возвращает наименьшее из двух значений.

№4

Рассмотрим число из I последовательности. Оно может принимать значение от $[-2^{16-1}; 2^{16-1}]$, если считать, что 0 и −0 — разные значения).

Итого от -2^{15} до 2^{15}. $[-2^{15}; 2^{15}]$

Число из II последовательности может принимать значение от -2^7 до 2^7 $[-2^7; 2^7]$

~~Так~~ Их произведение может принимать значения от $-2^7 \cdot 2^{15}$ до $2^7 \cdot 2^{15}$ $[-2^{22}; 2^{22}]$

Всего кол-во таких произведений равно $8 \cdot 16 = 2^3 \cdot 2^4 = 2^7$.

Тогда сумма 2^7 чисел от -2^{22} до 2^{22} может принимать значения от -2^{22+7} до 2^{22+7} — $[-2^{29}; 2^{29}]$

Для 1024 поданных пар будет выдано 1024 ответа которые могут принимать значения от $[-2^{29}$ до 2^{29}, т.е. 30-битные числа.

+10

Олимпиада школьников «Надежда энергетики»

Вариант: 37 III

ШИФР, НЕ ЗАПОЛНЯТЬ ⇨ RO51-27

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

$I = \frac{U}{R} \Rightarrow U = I \cdot R$

$\mathcal{E} = I(R + r)$

$\mathcal{E} = IR + Ir$

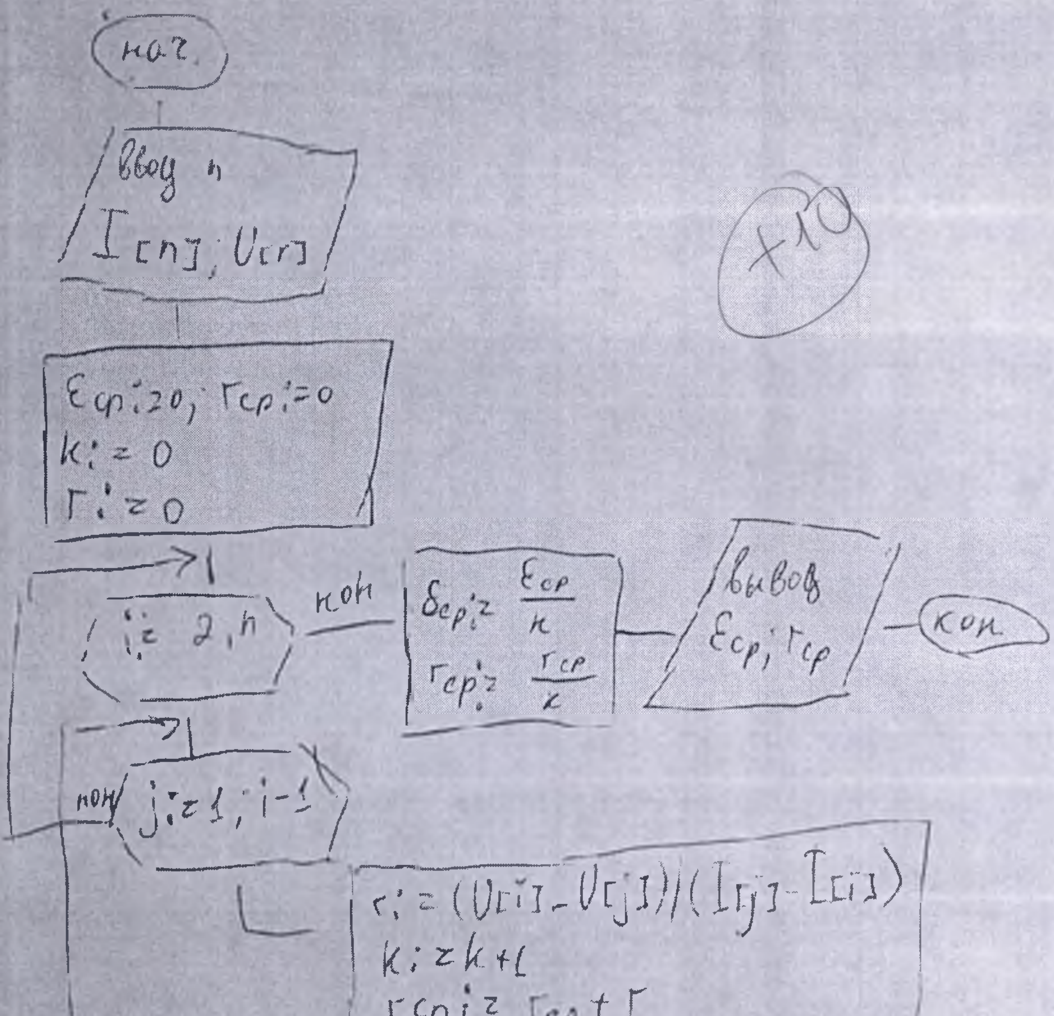
$\mathcal{E} = U + Ir$

$U_1 + I_1 r = U_2 + I_2 r$

$U_1 - U_2 = r(I_2 - I_1)$

$r = \frac{U_1 - U_2}{I_2 - I_1}$

+10



Олимпиада школьников «Надежда энергетики»

№ группы	Место проведения
I10F-01	Дистанционно, с использованием ВКС

QZ94-40

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 37101

ФАМИЛИЯ Шестаков

ИМЯ Вячеслав

ОТЧЕСТВО Григорьевич

Дата рождения 01.10.2005 Класс: 10

Предмет Информатика Этап: Заключительный

Работа выполнена на 5 листах Дата выполнения работы: [illegible]
(число, месяц, год)

Подпись участника олимпиады: [illegible]

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

Олимпиада школьников «Надежда энергетики»

Вариант: 37101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QZ94-40

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

Задача 1)

Пусть искомое число X_{10}. Тогда по свойству систем счисления: $X_{10} = 206_n = 2 \cdot n^2 + 0 \cdot n + 6 = 2n^2 + 6$. Т.к. в X_{10} 3 разряда, то: $100 \le X_{10} < 1000$

Представление X в 8-ричной и 13-ричной СС (системах счисления): $\overline{A0}_8$ и $\overline{B0}_{13}$, где A и B - группы цифр. Или же $\overline{A0}_8 = A \cdot 8 + 0 = X_{10} \Rightarrow$ $\Rightarrow X_{10} \vdots 8$ и $\overline{B0}_{13} = B \cdot 13 + 0 = X_{10} \Rightarrow X_{10} \vdots 13$. Поскольку ~~тогда~~ $X_{10} \vdots 8$ и $X_{10} \vdots 13$, то и $X_{10} \vdots НОК(8, 13)$, или же $X_{10} = k \cdot НОК(8,13) = k \cdot 104$. ~~Тогда~~

Теперь переберём все $X_{10} = 104k$ $(k \in N)$ из диапазона $100 \le X_{10} < 1000$ и проверим выполнимость $X_{10} = 2n^2 + 6$ (табл. 1.1). Давайте из последнего равенства выразим $n^2 = \frac{X_{10} - 6}{2}$ - это последний столбец таблицы. А теперь выпишем квадраты части натуральных чисел:

k	X_{10}	$X_{10} - 6$	$\frac{X_{10}-6}{2}$
≥ 10	≥ 1040	—	—
9	936	930	465
8	832	826	413
7	728	722	361
⋮	⋮	⋮	⋮

табл. 1.1

- $n \ge 22 \Rightarrow n^2 \ge 484$
- $n = 21 \Rightarrow n^2 = 441$
- $n = 20 \Rightarrow n^2 = 400$
- $n = 19 \Rightarrow n^2 = 361$

~~Видно, что при k =~~ Теперь смотрим в таблицу и видим:

- При $k \ge 10$ не выполняется, что $X_{10} < 1000$
- При $k = 9$, число $\frac{X_{10} - 6}{2} = 465$ - не является квадратом нат. числа
- При $k = 8$, число $\frac{X_{10} - 6}{2} = 413$ также не квадрат натурального числа
- При $k = 7$, число $\frac{X_{10} - 6}{2} = 361 = 19^2$ - подходит
- При $k \le 6$ будем иметь $\frac{X_{10} - 6}{2} < 361 \Rightarrow \sqrt{\frac{X_{10} - 6}{2}} < 19$, а значит если там и будут решения, то в них n ~~не~~ наибольшее.

Значит самое большее подходящее $n = 19$.

Ответ: 19.

x10

Олимпиада школьников «Надежда энергетики»

Вариант: 37101

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QZ94-40

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

Задача г)

Пусть длины труб это $a_1, a_2 \ldots a_n$. Давайте отсортируем числа так, чтобы $a_1 \le a_2 \le \ldots \le a_n$. Теперь будем перебирать варианты рекурсивно следующим образом.

- Пусть наша функция rec имеет двух параметра pos и sum
- Переберем все элементы $a_{pos}, a_{pos+1}, \ldots, a_{n-1}, a_n$
- Для каждого a_i запустим ~~X=rec(i+1)~~ (здесь i — индекс очередного перебираемого элемента) X=rec(i+1, sum−a_i)
- Если ~~в rec(i+1)~~ X равен −1, то продолжаем цикл
- ~~Если~~ Иначе возвращаем из функции число X+1 (поскольку мы смогли набрать сумму sum из a_i и каких-то других(!) чисел
- Если sum < 0, то возвращаем −1, если sum равен 0, возвращаем 0
- Если цикл закончился и ничего не нашлось — вернуть −1

Чтобы узнать требуемый в задаче ответ, нужно вызвать rec(0, M)

Обоснование верности алгоритма.

~~Если алгоритм выше вернул не оптимальный ответ, то значит что нечкая-то~~ Тогда Алгоритм выше выбрал набор чисел $b_1 \le b_2 \le \dots \le b_k$, где число считается "выбранным", если вариант с таким числом подошёл и вернулся из функции. Пусть оптимальный ответ $c_1 \le c_2 \le \dots \le c_\ell$. Тогда понятно, что если $c_1 < b_1$ — то такие варианты уже перебрались в нашем алгоритме. Значит $b_1 = c_1$, $b_2 = c_2$, ..., $b_k = c_k$, ~~...~~. Но тогда остались числа $c_{k+1}, c_{k+2}, \dots, c_\ell$. При этом $b_1 + b_2 + \dots + b_k = M$ и $c_1 + c_2 + \dots + c_\ell = M$. Значит из совокупности этих условий имеем $c_{k+1} + c_{k+2} + \dots + c_\ell = 0$, но такого не бывает, поскольку длины по условию натуральны.

То есть мы привели алгоритм (часть 1) и обосновали, что нет примеров оптимальнее (часть 2), а значит ответ, который вернётся приведенным выше алгоритмом и есть оптимальный. Возможно, что существуют алгоритмы, находящие ответ быстрее, но раз в задаче ничего не сказано про оптимизацию по времени, то этого делать не будем.

Олимпиада школьников «Надежда энергетики»

Вариант: 37101 ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ QZ94-40

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

Задача 3)

Пусть эксперементальные данные $a_1, a_2 \ldots a_n$. Будем сохранять усредненный ответ в $b_1, b_2, \ldots b_n$, усреднять будем средним арифметическим, числа b_i могут быть не целыми.
Заполним в следующем образом (если $n<2$, то $b_1=a_1$, иначе иначе):

- $b_1 = \frac{2a_1 + a_2}{3}$ – считаем, что до начала эксперемента было то же самое, что и во время его начала
- $b_n = \frac{a_{n-1} + 2a_n}{3}$ – считаем, что после конца эксперемента сохранились замеры конца эксперемента.
- Все другие $b_i = \frac{a_{i-1} + a_i + a_{i+1}}{3}$ $(2 \le i \le n-1)$

Теперь можно например, вывести в ~~[illegible]~~

Не очень понятно, что подразумевается фразой „провести очистку данных", поэтому на всякий случай придется привести код на C++:

```
#include <iostream>
using namespace std;
int main(){
  int n;
  cin >> n;
  int a[n];
  for(int &x: a) cin >> x; if(n<2){for(int x: a)cout << x << " "; return 0;}
  double b[n];
  b[0] = (2*a[0]+a[1])/3;
  b[n-1] = (2*a[n-1]+a[n-2])/3;
  for(int i=1; i<n-1; ++i) b[i] = (a[i-1]+a[i]+a[i+1])/3;
  for(auto x: b) cout << x << " ";
  return 0;
}
```

4/0

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

Задача 4)

Пусть первая последовательность – a, вторая – b. Тогда $0 \le a_i \le 2^{16}-1$; $0 \le b_i \le 2^8-1$. Тогда самое большое произведение $a_i b_j$ это $(2^{16}-1)(2^8-1) = 2^{24}-2^{16}-2^8+1 = X$. Понятно, что $2^{23} \le X < 2^{24}$. Значит на представление X нужно 24 бита. А минимальное произведение $a_i b_j = 0$, значит на хранение одного произведения меньше 24 бит тратить нельзя. ~~всего~~

Для одной пары наборов (a, b) считается $8 \cdot 16 = 2^7$ произведений. Значит сумма попарных произведений может быть $Y = 2^7 \cdot X = 2^{31}-2^{23}-2^{15}+2^7$. Очевидно, что $2^{30} \le Y < 2^{31}$, а минимальное Y бывает 0, значит ~~потреть~~ требуется по 31 биту на одну сумму. При этом если посчитать сумму Z ~~какого-то~~ всех произведений, кроме одного, то может получиться $Z = 2^{31}-2^{24}-2^{23}-2^{16}-2^{15}-2^8+2^7+1$, и также очевидно, что $2^{30} \le Z < 2^{31}$

x10

Рассмотрим конец подсчета сумм для всех наборов. В памяти хранится 1024 числа-суммы. До этого была совершена какая-то операция прибавления, а значит было 1023 таких же числа, 1 сумма была несколько меньше и еще где-то в памяти хранилось произведение, которое мы потом прибавили. Значит по рассуждениям в [illegible] потребуется 1024 "больших" числа (для сумм) и 1 "маленькое" число. В зависимости от толкования условия понятия "больших" и "маленьких" чисел несколько различаются:

- Можно работать с битами по отдельности, тогда будем хранить в начале памяти ~~сумму~~ произведение, а в конец добавлять ответы для наборов. Тогда нам понадобится 1 число ~~для текущ~~ длиной 24; 1024 числа длиной ~~23~~ 31 и одно число для индексации начала свободного участка памяти – 15 бит (т.к. всего нам нужно индексировать ~~24+2^{10}\cdot31~~ ~~2^{10}~~ $2^{10} \cdot 31 + 24$; $2^{14} < 2^{10} \cdot 31 + 24 < 2^{15}$)
- Чуть более реальный случай, когда мы работаем с байтами и ~~заун~~ каждое число должно занять целое число байт. Тогда также будем хранить в начале памяти произведение, а в конце накапливать суммы. Тогда понадобится 1 число длиной 24 бита = 3 байта, 1024 числа длиной 32 бита = 4 байта и одно число для индексации свободного участка памяти – 13 ~~байт~~ бит = 2 байта ($2^{12} < 1024 \cdot 4 + 3 < 2^{13}$)
- Еще могут быть запрещены числа длиной 3 байта, поэтому на это число (2-ой вариант) понадобится еще 1 байт.

Пример алгоритма, который будет расходовать ровно ~~тот~~ ~~столько~~ объём памяти, который необходимо, очевиден.

Олимпиада школьников «Надежда энергетики»

Вариант: 3710-1 ШИФР, НЕ ЗАПОЛНЯТЬ! QZ94-40

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

Рассматриваем все пары наборов по очереди

· Считаем все попарные произведения, сохраняем в эту [illegible]

· Прибавляем к нужной сумме

При этом где нужно бы хранить сами последовательности и перебираемых элементов, но фраза "сколько чисел (результатов)" позволяет это не делать. Также эта фраза говорит, что не нужна переменная, индексирующая свободное место в памяти. Значит:

Ответ $\begin{cases} \text{Числа занимают целое число бит — 1024? по 31 бит и 1? [illegible]} \\ \text{Числа занимают целое число байт — 1024? по 4 байта, 1 — 3 байта} \\ \text{Числа занимают четное число байт (не 3) — 1025? по 4 байта} \end{cases}$

Задача 5)

Будем считать, что $E = nB$, $r = 24$ Ом, а измеряются U и I. При этом измерения должны быть сделаны с одинаковой полярностью (U и I оба одного знака). Поскольку E и U — падения напряжения, то будем ~~[illegible]~~ проверять, чтобы $E = U$ с погрешностью $\le 1\%$. Также по закону Ома для полной цепи: $I = \frac{E}{R+r}$; [illegible], при этом на R накладывается физическое ограничение $R > 0$, т.е. $I = \frac{E}{R+r} \le \frac{E}{r}$

Составляем алгоритм (для пары U и I)

· Если $U \cdot I = 0$, то это ошибка в присоединении приборов

· Если $\frac{|E-U|}{E} > 0{,}01$ — то это ошибка в измерении напряжения

· Если $I > 1{,}01 \frac{E}{r}$ — то это ошибка в измерении силы тока ($R < 0$ не физично)

· Иначе данное измерение верно

Если нужно заключить вердикт относительно целой серии измерений, то нужно будет выполнить алгоритм выше для каждого замера и если найдется хоть одна ошибка, то ошибки есть, иначе — нет.

Если же при этом во время всех замеров сопротивление (внешнее) оставалось постоянным (хоть это и не понятно из условия!), то в конце еще нужно посчитать $R_i = \frac{U_i}{I_i} - r$ для каждого замера, найти $\langle R \rangle = \frac{\sum R_i}{n}$, и проверить, что $\frac{|R_i - \langle R \rangle|}{\langle R \rangle} < 0{,}01$ иначе имеется ошибка в одинаковости R.

+6

почти правильно!

здесь $IR = U$

Олимпиада школьников «Надежда энергетики»

ВСР

Место проведения

SG 24 25

шифр

← Не заполнять
Заполняется ответственным работником

Вариант № 3711

ФАМИЛИЯ ЩЕТИНИН

ИМЯ СТАНИСЛАВ

ОТЧЕСТВО ВЛАДИМИРОВИЧ

Дата рождения 20.12.2004

Класс: 11

Предмет ИНФОРМАТИКА

Этап: ЗАКЛЮЧИТЕЛЬНЫЙ

Работа выполнена на 04 листах

Дата выполнения работы: 20.02.2022
(число, месяц, год)

Подпись участника олимпиады: Щетинин

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.

```
for i=0...200:
    m = (Ri+Li)/2
    if ((U+Im)/(R+m) > I):
        Li=m
    else:
        Ri=m
```

$L_i = 0$
$R_i = 1000000$
Мы всегда берем среднее значение между L_i и R_i
200 операций хватит т.к. если ~~[illegible]~~ число операций $= k$, то само число 2^k

Теперь в переменной R_i находится r, то есть мы знаем r
Зная r мы можем найти $\mathcal{E}$
$\mathcal{E} = U + Ir$
Теперь на каждом эксперименте мы знаем r и $\mathcal{E}$
Если нам дано $2N$ наборов, то можно просто найти среднее арифметическое r и $\mathcal{E}$, как

$$\frac{r_1+r_2+\ldots r_{2N}}{2N} \quad и \quad \frac{\mathcal{E}_1+\mathcal{E}_2+\ldots \mathcal{E}_{2N}}{2N}$$

Не понятно, почему 200?!

Как $U+Im$?

Надо исп з-н Ома для полной цепи

+8

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

№4

Пусть дан массив a[8] – 16-ти разрядные числа, и массив b[16] – 8-ми разрядные числа, ans = 0.
Сделаем вложенные циклы. Сначала переберем i = 1…8 (номер числа из массива a), j = 1…16 (номер числа из массива b)
ans = ans + a[i] * b[j]
Тогда для одной пары последовательностей требуется хранить массив a, массив b и переменную ans
максимальное значение $ans = 16 \cdot 8 \cdot ((2^{16}-1)\cdot(2^{8}-1)) \approx$
$\approx 2^{7} \cdot 2^{16} \cdot 2^{8} = 2^{31}$ одна разряд – максимальное макс. знач. в массиве a, макс. знач. в массиве b
Тогда для ans нужно 32 бита (31 + 1 бит на знак)
Итого для одной пары нужна одна 32 битная переменная, 8 16-ти битных переменных, 16 8-ми битных переменных,
а для 1024 ~~пар~~ пар последовательностей нужно:
2^{10} 32 битных переменных, 2^{13} 16 битных, 2^{14} 8-ми битных
младший разряд

+8

№5

$I = \frac{\mathcal{E}}{R+r}$; $\mathcal{E} = IR + Ir$; $\mathcal{E} = U + Ir$; $U = \mathcal{E} - Ir$
известно известно

I – сила тока, $\mathcal{E}$ – ЭДС, R – сопротивление нагрузки,
U – напряжение, r – внутреннее сопротивление
$U = IR \Rightarrow R = \frac{U}{I} \Rightarrow$ при каждом эксперименте мы можем узнать R.

$I = \frac{U + Ir}{R + r} \Rightarrow$ мы можем найти r
известно, известно, известно

Для этого применим бинарный поиск, но нужно доказать, что $I(r) = \frac{\mathcal{E}}{R+r}$ – монотонная возрастающая (убывающая) функция. Возьмем производную: $I'(r) = \frac{\mathcal{E}'(R+r) - (R+r)'\mathcal{E}}{(R+r)^2} =$
$= -\frac{\mathcal{E}}{(R+r)^2}$, $-\frac{\mathcal{E}}{(R+r)^2} \geq 0$, т.к. $(R+r)^2 \geq 0$, то $\mathcal{E} \leq 0 \Rightarrow$
функция убывающая при $r \in R$, значит по ней можно запустить бинарный поиск; сделаем 200 операций и найдем значение максимально приближенное к r (200 точно будет достаточно)

и из всех значений найти большее. Пусть будут также переменные ans = 0, ans1 = 0, ans2 = 0, ans3 = 0

Внутри трех вложенных циклов посчитаем такое:

d1 = sqrt((fr[x]-sc[x])*(fr[x]-sc[x])+(fr[y]-sc[y])*(fr[y]-sc[y])+(fr[z]-sc[z])*(fr[z]-sc[z]))

d2 = sqrt((sc[x]-th[x])*(sc[x]-th[x])+(sc[y]-th[y])*(sc[y]-th[y])+(sc[z]-th[z])*(sc[z]-th[z]))

d3 = sqrt((fr[x]-th[x])*(fr[x]-th[x])+(fr[y]-th[y])*(fr[y]-th[y])+(fr[z]-th[z])*(fr[z]-th[z]))

$P = d_1 + d_2 + d_3$

Надо проверить существует ли Δ?

x7

Делаем проверку: если P > ans, тогда ans = P, ans1 = fr, ans2 = SC, ans3 = th

По окончании циклов выводим ans1, ans2, ans3 - треугольник с наиб. периметром

N3

Дан массив a[N][N]. Создадим массив ans[N][N]

Вложенными циклами переберём координаты каждой ячейки:

i = 1..N (номер строки) j = 1..N (номер столбца)

x10

Этим циклом мы перебираем ячейку. Сделаем ~~внутри~~ внутри еще два вложенных цикла: x = max(0, i-1) .. min(N, i+1)
y = max(0, j-1) .. min(N, j+1) - он будет верно обрабатывать случаи (1), (2), (3)

Будем поддерживать переменную cnt (=0 изначально) и делать так:

```
for x = ...
  for y = ...
    ans[i][j] = ans[i][j] + a[x][y]
    cnt = cnt + 1
```

По завершении циклов с x и y сделаем:

ans[i][j] = ans[i][j] / cnt и тогда в ans[i][j] будет храниться среднее по соседним элементам ~~параметрам~~

Ответ: массив ans.

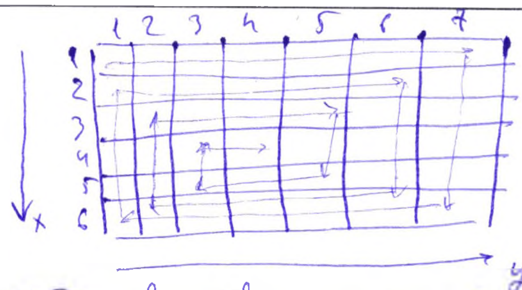
Олимпиада школьников «Надежда энергетики»

Вариант: 37111

ШИФР, НЕ ЗАПОЛНЯТЬ! ⇨ SG 24-25

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

№1 Для начала заполним всю таблицу нулями. Возьмем координаты начала $x=1$, $y=1$ и направление $g=1$ (пусть, если нужно идти вправо, тогда $g=1$, влево: $g=2$, вниз: $g=3$, вверх: $g=4$), также $i=1$ (само побуральное число) и запишем его в координату $(x;y)$



Напишем такой алгоритм:

1) Если $g=1$ и $(x;y+1)=0$ и $y<M$ (M - кол-во столбцов), тогда увеличиваем y на 1 и в клетку $(x;y)$ записываем i, предварительно увеличенное на 1. Иначе $g=3$

2) Если $g=3$ и $x<N$ и $(x+1;y)=0$, тогда увеличиваем x на 1 и в клетку $(x;y)$ записываем i, предварительно увеличенное на 1. Иначе $g=2$

3) Если $g=2$ и $y>1$ и $(x;y-1)=0$, тогда уменьшаем y на 1 и в клетку $(x;y)$ записываем i, предварительно увеличенное на 1. Иначе $g=4$

4) Если $g=4$ и $x>1$ и $(x-1;y)=0$, тогда уменьшаем x на 1 и в клетку $(x;y)$ записываем i, предварительно увеличенное на 1. Иначе $g=1$

Все эти 4 цикла должны находиться внутри одного большого, который прекратит свою работу, если мы не выполнили ни одну операцию в любом из циклов. Для этого можно хранить переменную типа bool, которая изначально равна true, перед началом работы внутренних циклов сделаем её равной false и если мы вошли в один из 4 циклов сделаем её true. Если по итогу у нас значение true, то продолжаем выполнять цикл, иначе прерываем программу и выводим таблицу

№2

Сделаем три вложенных цикла. В первом $i=1\ldots N$ (i - номер строки первой точки), второй $j=i+1\ldots N$ (j - номер строки второй точки), третий $k=j+1\ldots N$ (k - номер строки третьей точки). Таким образом, $i\neq j\neq k$, но при этом перебираются все случаи. Пусть $fr=a[i]$, $sc=a[j]$, $th=a[k]$. Тогда нам нужно узнать: (расстояние от fr до sc) + (расстояние от sc до th) + (расстояние от th до fr)