Разбор задач

Задача 1. Длинный плакат

Ответ:

```
5867
86912
9988887695
999956789123456789123456789123456789
988776655432143210321021010
```

В первом примере из числа 43521867 нужно вычеркнуть 4 цифры. То есть первой цифрой может стать любая цифра из первых пяти: 43521. Чтобы число получилось максимальным, нужно оставить наибольшую возможную цифру из этих. Это цифра 5, поэтому нужно вычернуть первые две цифры. Получилось ~~43~~**5**21867. Осталось вычернуть ещё 2 цифры после цифры 5, то есть можно вычернуть 21 и получится число ~~43~~5~~21~~867.

Во втором примере нужно вычернуть пять цифр из числа 7854635912, «добраться» до девятки не удаётся, поэтому вычеркнем цифры до восьмёрки: ~~7~~**8**54635912. Осталось вычернуть четыре цифры, из пяти следующих цифр 54635, максимальная — это 6. Получаем ~~7~~**8**~~54~~**6**35912. Осталось вычернуть две цифры, можно вычеркуть 35 и оставить девятку: ~~7~~**8**~~54~~**6**~~35~~**9**12, все пять цифр уже вычеркнуты, оставшиеся цифры 12 добавляются к ответу.

Аналогично строятся ответы и для оставшихся трёх заданий.

Задача 2. Танец

Ответ:

```
8
56
600
266
820
```

Ответ для первой расстановки можно получить, промоделировав описанный в условии процесс.

Ответ для второго примера проще посчитать, если заметить, что каждая девочка должна поменяться местами с каждым мальчиком, стоящим левее неё. То есть каждая из 8 девочек поучаствует в 7 обменах и общее число обменов равно $8 \times 7 = 56$.

В третьем примере 10 девочек делают по 10 обменов, 10 девочек — по 20 обменов, 10 девочек — по 30 обменов, общее число обменов равно $10 \times 10 + 10 \times 20 + 10 \times 30 = 600$.

В четвёртом примере ответ равен

$$1\times1+2\times(1+2)+3\times(1+2+3)+4\times(1+2+3+4)+5\times(1+2+3+4+5)+6\times(1+2+3+4+5+6) = 266.$$

Эту сумму можно также посчитать на бумагие или при помощи приложений на компьютере.

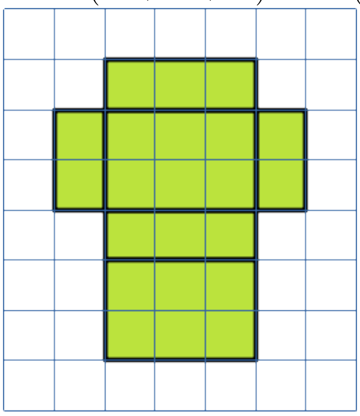
В пятом примере ответ равен $1 + 2 + 3 + ... + 40$. Это арифметическая прогрессия, сумма которой равна $\frac{1+40}{2} \times 40 = 820$. Незнакомые с соответствующей формулой могут посчитать эту сумму по методу, приписываемому Карлу Гауссу: разобъём слагаемые на 20 пар с суммой 41: $(1 + 40) + (2 + 39) + ... + (20 + 21) = 41 \times 20 = 820$. Также можно было использовать компьютер: например, заполнить электронную таблицу числами от 1 до 40 и посчитать сумму чисел в блоке.

Задача 3. Развёртка параллелепипеда

Ответ:

$$8 * a + 4 * b + 2 * c$$

У параллелепипеда 4 ребра длины a, 4 ребра длины b и 4 ребра длины c. Если же посчитать сумму периметров всех граней, то получится вдвое больше, так как одно ребро принадлежит двум граням. Периметр развёртки будет равен этой сумме, за вычетом тех рёбер, которые находятся внутри развёртки, то есть которые соединяют прямоугольники в развёртке, при этом каждое такое ребро также нужно вычесть два раза. Всего в развёртке шесть прямоугольников, они соединены между собой пятью рёбрами. Чтобы получить минимальный периметр развёртки, нужно соединять прямоугольники при помощи самых длинных рёбер (длины c). Использовать все четыре ребра длины c для соединения прямоугольников внутри развёртки не получится (такой развёртки не существует), но можно использовать три ребра длины c. Оставшиеся два ребра можно взять длины b. Получится ответ $2 \times (4a + 4b + 4c) - 2 \times (3a + 2b) = 8a + 4b + 2c$. Такая развёртка существует, вот пример.



Задача 4. Парковка

Можно разместить 28 машин, вот пример такого размещения:

```
..AAAAA
A......
A.AAAAA
A.AA..A
A.AAA.A
A.....A
.AAAAA.
```

Задача 5. Сдача

60 баллов можно набрать моделированием процесса выдачи сдачи по одному человеку, пока не кончатся монеты по одному рублю. Выдавая сдачу прежде всего нужно потратить двухрублёвые монеты, их понадобится $\lfloor \frac{n}{2} \rfloor$ для каждого пассажира, но также нужно учитывать, что у нас не более b монет по два рубля (значение b будем уменьшать по мере выдачи сдачи). Оставшуюся часть сдачи выдадим рублёвыми монетами. Будем продолжать этот процесс, пока число рублёвых монет (переменная a) не станет отрицательным — это означает, что последнему пассажиру сдачу выдать не удалось. Такое решение не будет укладываться в ограничение по времени при больших входных числах. Пример такого решения на языке Python.

```python
n = int(input())
a = int(input())
b = int(input())
ans = 0
while a >= 0:
    count2 = min(b, n // 2)
    count1 = n - 2 * count2
    a -= count1
    b -= count2
    ans += 1
print(ans - 1)
```

Чтобы набрать полный балл, посчитаем общую сумму денег у кондуктора, она равна $s = a + 2b$. Количество пассажиров не может быть больше $\lfloor \frac{s}{n} \rfloor$, иначе просто не хватит денег, но также следует учесть, что если n нечётное, то каждому пассажиру нужно выдать минимум одну рублёвую монету, поэтому при нечётном n ответ не может быть больше a и нужно взять минимальное из чисел a и $\lfloor \frac{s}{n} \rfloor$. Пример решения на 100 баллов на языке Python.

```
n = int(input())
a = int(input())
b = int(input())
s = a + 2 * b
ans = s // n
if n % 2 != 0:
    ans = min(ans, a)
print(ans)
```

Задача 6. BoxStation

Чтобы набрать 60 баллов, можно промоделировать процесс по дням. Пусть s — количество приставок в магазине, а $delta$ — на сколько увеличивается это количество каждый день. В самом начале $delta = m - n$, затем каждый день $delta$ уменьшается на 1, а значение s увеличивается на $delta$. Рано или поздно $delta$ станет отрицательным и значение s начнёт уменьшаться. Цикл закончится, когда s станет отрицательным, нужно вывести количество итераций цикла (количество прошедших дней), которое будем сохранять в переменной day. Пример решения на языке Python.

```
n = int(input())
m = int(input())
s = 0
delta = m - n
day = 0
while s >= 0:
    s += delta
    delta -= 1
    day += 1
print(day)
```

Чтобы решить эту задачу на полный балл, нужно заметить, что процесс накопления приставок в магазине и процесс уменьшения числа накопленных приставок симметричны. Пусть $n < m$, тогда первые $m - n$ дней количество приставок в магазине растёт: в первый день на $m - n$, во второй — на $m - n - 1$ и т.д. В день номер $m - n$ количество приставок в магазине увеличилось на одну, на следующий день количество приставок не изменилось (привезено было столько же, сколько продали), затем количество приставок уменьшается: сначала на 1, потом на 2 и т.д. Уменьшение будет происходить также $m - n$ дней, поэтому через $(m - n) + 1 + (m - n)$ дней в магазине останется ноль приставок, и на следующий день приставок не хватит.

Поэтому ответ равен $2(m-n)+2$ кроме случая, когда $n > m$, в этом случае ответ равен 1, потому что уже в первый день продаж приставок не хватит.

Пример решения на языке Python.

```
n = int(input())
m = int(input())
print(max(1, 2 * (m - n) + 2))
```

Задача 7. Лес

Чтобы набрать 60 баллов, можно промоделировать весь путь Миши по одному метру. Если координаты Миши (x, y), то он вышел из леса, когда $|x| = K$ или $|y| = K$.

Закодируем направления движения числами 0 (север), 1 (восток), 2 (юг), 3 (запад). В массивах `DX` и `DY` хранятся величины перемещения на один метр в данном направлении, то есть если направление движения есть `dir`, то по оси OX Миша сместится на значение `DX[dir]`, а по оси OY — на значение `DY[dir]`. Миша меняет направления движения в порядке 0, 1, 2, 3, 0, 1, 2, 3. Если пронумеровать отрезки пути Миши числами от 0 до $n - 1$, то тогда на i-м отрезке направление его движения есть $i \bmod 4$ (остаток от деления i на 4).

Внутри цикла по всем отрезкам будем менять значение координат (x, y) прибавляя к ним `DX[i % 4]` и `DY[i % 4]`. Когда Миша окажется на границе леса, выведем координаты текущей точки и завершим работу программы.

```
import sys

DX = [0, 1, 0, -1]
DY = [1, 0, -1, 0]
k = int(input())
n = int(input())
x = 0
y = 0
for i in range(n):
    a = int(input())
    for j in range(a):
        x += DX[i % 4]
        y += DY[i % 4]
        if abs(x) == k or abs(y) == k:
            print(x, y)
            sys.exit(0)
print(x, y)
```

Это решение можно улучшить, если моделировать перемещения не по одному метру, а сразу же на весь отрезок длины `a` метров. В этом случае к значению x будет добавляться `DX[i%4] * a`, к значению y будет добавляться `DY[i%4] * a`. При таком моделировании, после очередного перемещения Миша может выйти за границы квадрата, поэтому цикл продолжается, пока обе координаты по модулю меньше k. Если же очередной конец отрезка оказался вне квадрата, то нужно вычесть излишне пройденные вне квадрата метры (то есть если Миша оказался в точке (x, y) и $|x| > k$, то нужно вернуться на $|x| - k$ излишне пройденных метров, если же $|y| > k$, то нужно вернуться на $|y| - k$ метров). Пример решения на 100 баллов.

```
import sys

DX = [0, 1, 0, -1]
DY = [1, 0, -1, 0]
k = int(input())
n = int(input())
x = 0
y = 0
for i in range(n):
    a = int(input())
    x += DX[i % 4] * a
    y += DY[i % 4] * a
    if abs(x) >= k:
        x -= (abs(x) - k) * DX[i % 4]
        break
    if abs(y) >= k:
        y -= (abs(y) - k) * DY[i % 4]
        break
print(x, y)
```