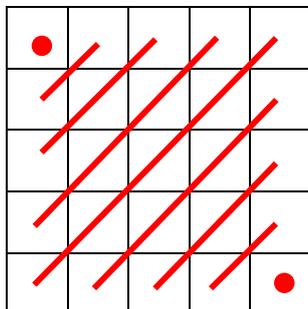


Задача А. Табличная змейка



1	2	6	7	15
3	5	8	14	16
4	9	13	17	22
10	12	18	21	23
11	19	20	24	25

Можно заметить, что в каждой «диагонали» змейки количество чисел увеличивается на 1 (пока не дойдем до побочной диагонали таблицы, где и расположены верхняя правая и нижняя левая ячейки).

Поэтому, до того, как начнем заполнять побочную диагональ, в таблицу уже будут внесено количество чисел:

$$1 + 2 + 3 + \dots + 99 = \frac{1 + 99}{2} * 99 = 4950.$$

В побочной диагонали матрицы будет 100 чисел, начиная от 4951 до 5050. По траектории заполнения можно заметить, что элементы в «диагоналях», начинающихся на строках с четным номером, числа расположены по убыванию (если читать слева-направо). Поэтому в левом нижнем углу будет стоять большее число – 5050, а в правом верхнем – 4951.

Задача В. Зеркальные числа

Ответ:

121314	141
18237465	8
56738306205	653080356
42349763929424	94432923449

Задача С. Перерыв на лекции

Задача является арифметической.

Исходя из условия, следует вычислить время середины лекции и «отложить» в обе стороны от середины время, равное половине перерыва.

Если длительность лекции или длительность перерыва не делятся нацело на два, то следует учесть, что длительность первой части должна быть больше.

Пример программы на языке Pascal:

```
var
  a, b, c, d, m, start, fin, dlit, middle, beg: integer;

begin
  read(a, b, c, d, m);
  start := 60 * a + b;
  fin := 60 * c + d;
  dlit := fin - start;
  if dlit mod 2 = 0 then
  begin
    middle := start + dlit div 2;
    beg := middle - m div 2;
  end
  else
  if m mod 2 <> 0 then
    beg := start + (dlit - m) div 2
  else
    begin
      middle := start + (dlit + 1) div 2;
      beg := middle - m div 2;
    end;
  write(beg div 60, ' ', beg mod 60)
end.
```

Задача D. Раздел клада

Заметим, что если монет в мешке 1, 2 или 3, то Альк заберет весь клад. Если монет 4, то Альк проиграет, так как сколько бы монет он не взял, Бен одним ходом заберёт остальные. Если монет 5, 6 или 7, то Альк может взять 1, 2 или 3 монеты соответственно, в мешке останется 4 монеты, и Бен проиграет, а Альк выиграет.

Анализируя аналогичным образом большие значения n , можно заметить закономерность, что Альк проигрывает только в случае, когда $n : 4$, а во всех остальных – выигрывает.

Пример программы на языке Pascal

```
var n: integer;
begin
  read(n);
  if n mod 4 = 0 then
    write('Ben')
  else
    write('Alk');
end.
```

Задача E. Считалочка

С учетом имеющихся ограничений на N и K данная задача может быть решена моделированием данного процесса.

Создадим массив, в котором будем хранить 1, если участник с соответствующим номером еще не выбыл, и 0 – в противном случае.

Далее будем вычислять, кто из участников будет удален на следующем круге считалки, и заменять соответствующий элемент нулем (с тех пор, пока в массиве не остается единственный ненулевой элемент).

Пример программы на языке Pascal

```
var
  n, k, i, j, d: integer;
  a: array[0..1000] of byte;

begin
  read(n, k);
  for i := 0 to n - 1 do
    a[i] := 1;
  d := 0;
  i := 0;
  j := 0;
  while d <> n - 1 do
    begin
      if a[i] = 1 then
        begin
          j := j + 1;
          if j = k then
            begin
              a[i] := 0;
              j := 0;
              d := d + 1;
            end;
          end;
        i := (i + 1) mod n;
      end;
      i := 0;
      while a[i] = 0 do
        i := i + 1;
      write(i + 1);
    end.
```