

7-8 класс

1. «Детская задачка» (100 баллов)

Эта задача не требует написания программы. Необходимо только вывести формулу. Пусть у Тамары x конфет. Тогда у Вики $x + b$. Так как вместе у них a конфет, то получим уравнение

$$x + (x + b) = a$$

$$x = (a - b) / 2$$

Ответ: $(a - b) / 2$

В качестве ответа будет засчитана любая другая эквивалентная формула.

2. «Вписываемся в бюджет» (100 баллов)

Ответы к данной задаче содержатся в файлах:

zarplata_cor01.txt

9259

zarplata_cor02.txt

15151

Эта задача не требует написания программы, но прилагается и программная реализация решения данной задачи в файле **z2_zarplata.pas**.

Также в файле **z2.xlsx** предлагается решение данной задачи в MS Excel. Для первого теста решение находится на листе 2А, для второго теста – на листе 2В. Данные из входных файлов скопированы на лист. Скопируем соответствующие таблицы из текстовых файлов. В отдельную ячейку D3 введем любое число – предполагаемый оклад уборщика. В столбце D далее посчитаем зарплату для каждой должности. В ячейке E3 найдем суммы зарплат. Для удобства в ячейку F3 введем формулу =ЕСЛИ(E3>A1;1;0). Теперь легко подобрать максимальный оклад уборщика так, чтобы сумма зарплат не превосходила зарплатный фонд. Неважно, каким способом участник получил ответ: если ответ удовлетворяет условию задачи, то баллы начисляются.

3. «Шашки» (100 баллов)

Ответ к данной задаче содержится в файлах:

shaska_cor01.txt

4 2 1 3 5 4 2 3

shashka_cor02.txt

3 5 6 4 2 1 3 5 7 6 4 2 3 5 4

Необходимо написать алгоритм в соответствующей форме. Если форма записи не соответствует условию (например, алгоритм начинается с перемещения черной шашки), то баллы не начисляются. Для каждой подзадачи 25 баллов будет начислено за алгоритм, который приводит к правильному ответу, и еще 25 баллов за алгоритм с минимальным количеством команд (8 команд для теста А и 15 команд для теста Б).

4. «Страусы и жирафы» (100 баллов)

```
var a, b, s, g: integer;
begin
  readln(a, b);
  if (a mod 2 <> 0) or (b mod 2 <> 0) then
  begin
    writeln(-1, ' ', -1);
  end
  else
  begin
    s := a - b div 2;
    g := (b - a) div 2;
    if (s < 0) or (g < 0) then
    begin
      writeln(-1, ' ', -1);
    end
    else
    begin
```

```

        writeln(s, ' ', g);
    end;
end;
end.

```

Обозначим количество страусов s , а количество жирафов g . Так как у каждого животного два глаза, то $2s + 2g = a$. Из того, что у страуса две ноги, а у жирафа 4, следует, что $2s + 4g = b$.

Решим полученную систему уравнений. Умножим первое уравнение на 2:

$$\begin{cases} 4s + 4g = 2a \\ 2s + 4g = b \end{cases}$$

Вычтем из первого уравнения второе: $2s = 2a - b$. Откуда $s = a - \frac{b}{2}$.

Теперь из первоначального второго уравнения вычтем первое: $2g = b - a$. Откуда

$$g = \frac{b - a}{2}.$$

Найти количество животных нельзя, если a или b нечетное, или найденные значения s или g получились отрицательные.

5. «Тигренок» (100 баллов)

```

var x, y: double;
    p, d: integer;
begin
    readln(x, p, y);
    d := 3;
    while x < y do
    begin
        x := x + x * p / 100;
        d := d + 1;
    end;
    writeln(d, ' ', x);
end.

```

Задача на моделирование ситуации, описанной в условии задачи. Необходимо просто написать цикл и в нем последовательно увеличивать переменную, в которой хранится возраст (в месяцах) тигренка. Необходимо только не забыть, что впервые мясо тигренку дали в 3 месяца, а значит, это и будет начальным значением переменной.

6. «Катание на лифте» (100 баллов)

```

var n, m, i, j, x1, x2: integer;
    name: string;
    names_all: array of string;
    is_up: boolean;
begin
    readln(n);
    for i := 1 to n do
    begin
        readln(name);
        readln(m);
        readln(x1);
        is_up := true;
        for j := 1 to m - 1 do
        begin
            readln(x2);
            if x1 > x2 then
                is_up := false;
            x1 := x2;
        end;
        if is_up then
            begin

```

```
        setlength(names_all, length(names_all) + 1);
        names_all[length(names_all) - 1] := name;
    end;
end;
for i := 0 to length(names_all) - 1 do
begin
    writeln(names_all[i]);
end;
end.
```

В задаче требуется среди всех последовательностей чисел (этажи, на которых останавливался каждый из детей) выбрать те, которые являются возрастающими. При решении можно записывать каждую последовательность в отдельный массив, а можно это делать по ходу считывания данных, как приведено в авторском решении.

