

### Задача А. Метеооповещения

Данная задача является самой простой в данном туре. Для её решения необходимо просто проверить три простых условия в нужном порядке и вывести необходимые сообщения. Пример фрагмента кода на языке С, который решает задачу, показан ниже:

```
int t1, t2, v1, v2;
scanf("%d%d", &t1, &v1);
scanf("%d%d", &t2, &v2);

if ((t2 < 0) && (v2 >= 10)) {
    printf("A storm warning for tomorrow! Be careful and stay home if possible!");
} else if (t2 < t1) {
    printf("MCHS warns! Low temperature is expected tomorrow.");
} else if (v2 > v1) {
    printf("MCHS warns! Strong wind is expected tomorrow.");
} else {
    printf("No message");
}
```

### Задача В. Безналичный перевод

Для решения этой задачи необходимо рассчитать размер комиссии и после этого проверить полученный результат, что бы он не был меньше 100 и не больше 2000. После этого необходимо вывести результат в нужном формате. При этом при желании можно обойтись даже без вещественных чисел. Пример фрагмента кода на языке С показан ниже:

```
int x, percent;
scanf("%d", &x);

percent = x + 2500;
if (percent < 10000) {
    percent = 10000;
}
if (percent > 200000) {
    percent = 200000;
}

printf("%.02d\n", percent / 100 , percent % 100);
```

### Задача С. Загадочное уравнение

Постановка задачи. Дано уравнение  $x + y + xy = n$ . Найти количество пар целых неотрицательных чисел  $x$  и  $y$ , которые являются решением этого уравнения.

Идея решения. Заметим, что многочлен  $x + y + xy$  симметричен относительно своих переменных. Это значит, что если какая-то пара  $x = a$  и  $y = b$  является решением уравнения, то пара  $x = b$  и  $y = a$  также является решением данного уравнения. Тогда, подобно алгоритму проверки числа на простоту, переберем все  $x$  не превосходящие  $\sqrt{n}$ .

Для каждого  $x$  восстановим  $y$ :  $y = (n - x) / (x + 1)$ . Если  $y$  целое, добавим пары  $(x, y)$  и  $(y, x)$  в ответ.

Не забудем, что если  $x = y$ , то нужно добавить только одну пару.

### Задача D. К-перестановки

Заметим, что перестановок из девяти чисел немного, всего  $9! = 362880$ . Поэтому можно просто перебрать все перестановки из  $n$  чисел и проверить каждую, не является ли она  $k$ -перестановкой.

Приведем фрагмента программы на языке Паскаль.

```
procedure go(pos : integer);
var
  i : integer;
begin
  if (pos = n + 1) then begin
    for i := 1 to n - 1 do begin
      if abs(p[i] - p[i + 1]) > k then begin
        exit;
      end;
    end;
    inc(ans);
    exit;
  end;
  for i := 1 to n do begin
    if (not w[i]) then begin
      w[i] := true;
      p[pos] := i;
      go(pos + 1);
      w[i] := false;
    end;
  end;
end;
```

Процедура `go(pos)` перебирает все возможные еще не использованные числа, которые можно поставить на позицию `pos` в перестановке `p`, если мы знаем все предыдущие числа. Для того, чтобы эффективно проверять, было ли использовано число или нет, поддерживается булевый массив `w`, причем `w[i] = true` если `i` уже использовано и `false` иначе. Если `pos > n`, то мы построили какую-то перестановку, осталось проверить, что она является `k` - перестановкой. Если это так, то увеличиваем ответ переменную `ans`.