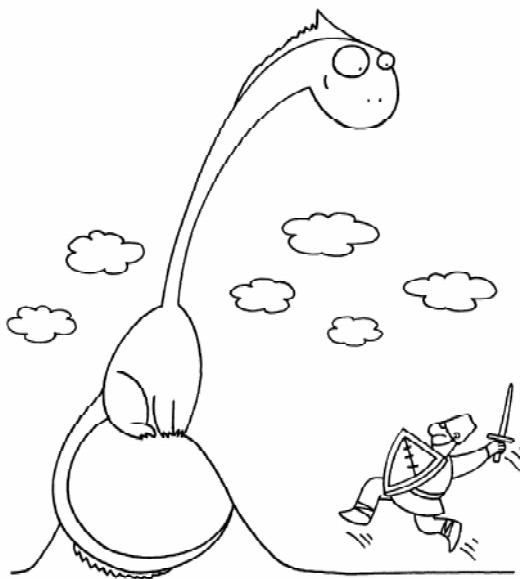


*Комаров Андрей Валерьевич,
Ульянцев Владимир Игоревич,
Цыпленков Алексей Евгеньевич*

ЗАДАЧА «РЫЦАРСКИЙ ЩИТ»

Этой статьей мы продолжаем цикл публикаций олимпиадных задач для школьников по информатике. Решение таких задач и изучение разборов поможет Вам повысить уровень практических навыков программирования и подготовиться к олимпиадам по информатике.

В этой статье рассматривается задача «Рыцарский щит», которая предлагалась на одиннадцатой всероссийской командной олимпиаде школьников по программированию (ВКОШП). Сайт этой олимпиады находится по адресу <http://neerc.ifmo.ru/school/russia-team/>.



УСЛОВИЕ ЗАДАЧИ

Погостив пару недель у Темного Властелина и прослушав истории о всех его похождениях за последние годы, сэр Петрейн понял, что он уже давно не совершал никаких подвигов. Посидев за чашкой чая и тщательно обсудив будущий подвиг, они решили, что Петрейну нужно победить ужасного дракона, который уже давно терроризирует западные окраины королевства. И вот он отправился готовиться к великому походу.

Но какой рыцарь идет на дракона без рыцарского обмундирования? Петрейну нужны доспехи, щит и меч. Доспехи и меч у него есть, а щита нет. Всем известно, что, чем щит больше, тем эффективней будет он в бою. Сейчас у Петрейна есть два треугольных щита, но он считает их недостаточно надежными и хочет сделать из них один.

Королевский оружейник, взявшийся за изготовление щита, предложил следующий способ: два имеющихся щита кладутся рядом так, чтобы они соприкасались сторонами и фиксируются в таком положении. Сэр Петрейн заметил, что как бы оружейник ни старался, у полученного в результате щита всегда будет одинаковая площадь, а значит, его эффективность в бою с драконом будет зависеть

только от качества щитов, но не от способа их скрепления.

Но ему нужен не просто кусок металла, а щит с символикой его рода: золотым обрамлением по периметру. Однако золото стоит дорого, поэтому Петрейну хочется, чтобы периметр полученного щита был как можно меньше. Помогите ему выяснить, какой минимальный периметр может иметь полученный щит.

Формат входного файла

В первой строке заданы три числа a_1 , b_1 и c_1 – длины сторон первого щита. Во второй строке заданы три числа a_2 , b_2 и c_2 – длины сторон второго щита. Обе строки задают корректные невырожденные треугольники. Все числа во входном файле целые и не превосходят 100000.

Формат выходного файла

Выведите единственное число – минимальный периметр щита, который можно изготовить из заданных треугольных щитов указанным способом.

Примеры входных и выходных данных

shield.in	shield.out
1 1 1 1 1 1	4
3 4 5 8 7 6	23

РАЗБОР ЗАДАЧИ

Рассмотрим заданные треугольники. Пусть P_1 и P_2 – периметры первого и второго треугольников соответственно. Рассмотрим периметр P фигуры, заданной

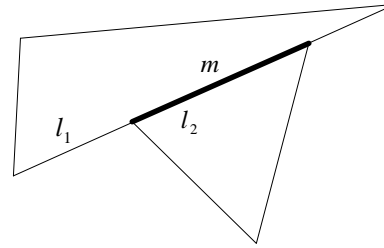


Рис. 1. Длина m общей части сторон с длинами l_1 и l_2

описанным в условии способом. Пусть длина соприкосновения треугольников равна m . Тогда полученная фигура будет иметь периметр $P = P_1 + P_2 - 2m$. Ответом на задачу является наименьшее возможное значение P . Так как величины P_1 и P_2 постоянны, то P будет зависеть только от значения m . Следовательно, необходимо найти m_{\max} – наибольшее значение m .

Пусть стороны, которыми соприкасаются треугольники, имеют длины l_1 и l_2 . Тогда наибольшая длина их общей части m , как показано на рис. 1, равна $\min(l_1, l_2)$.

Применив данный факт, можно решить задачу следующим образом. Переберём все пары чисел l_1 и l_2 такие, что l_1 является длиной стороны первого треугольника, а l_2 – второго. Для каждой такой пары считаем величину $\min(l_1, l_2)$ и выберем максимальное её значение m_{\max} по всем парам. Тогда наименьшее значение периметра P будет равно $P_1 + P_2 - 2m_{\max}$. Реализация данного решения продемонстрирована в листинге 1.

Теперь заметим, что если $a \leq b$, то $\min(a, c) \leq \min(b, c)$. Это позволяет найти m_{\max} по формуле $m_{\max} = \min(\max(a_1, b_1, c_1), \max(a_2, b_2, c_2))$.

Листинг 1. Реализация предложенного алгоритма

```
uses
    Math;

var
    first, second : array [1..3] of longint;
    i, j : longint;
    ans : longint;

function calc(i, j : longint) : longint;
var
    k : longint;
```

```
begin
  result := 0;
  for k := 1 to 3 do begin
    inc(result, first[k]);
    inc(result, second[k]);
  end;
  result := result - 2 * min(first[i], second[j])
end;
begin
  reset(input, "shield.in");
  rewrite(output, "shield.out");
  read(first[1], first[2], first[3]);
  read(second[1], second[2], second[3]);

  ans := 1000000;
  for i := 1 to 3 do begin
    for j := 1 to 3 do begin
      ans := min(ans, calc(i, j));
    end;
  end;
  writeln(ans);
end.
```

В листинге 2 приведена реализация алгоритма с учётом данного замечания.

Время работы алгоритма не зависит от величин, содержащихся во входном файле, и составляет $O(1)$.

Листинг 2. Реализация улучшенного алгоритма

```
ans := 0;

for i := 1 to 3 do begin
  ans := ans + first[i];
  ans := ans + second[i];
end;

max1 := -1;
max2 := -1;

for i := 1 to 3 do begin
  max1 := max(max1, first[i]);
  max2 := max(max2, second[i]);
end;

writeln(ans - 2 * min(max1, max2));
```

Члены жюри Интернет-олимпиад по информатике:

*Комаров Андрей Валерьевич,
студент первого курса кафедры КТ
СПбГУ ИТМО, призёр всероссийской
олимпиады по информатике,*

*Ульянцев Владимир Игоревич,
студент четвертого курса кафедры
КТ СПбГУ ИТМО, член жюри
ВКОШП.*

*Цыпленков Алексей Евгеньевич,
студент второго курса кафедры КТ
СПбГУ ИТМО*



Наши авторы, 2010.

Our authors, 2010.