

# 1. Задача

Исходные данные:

Программа производит чтение трех целых чисел, которые интерпретируются как длины сторон треугольника. Далее программа печатает сообщение о том, является ли треугольник неравносторонним, равнобедренным или равносторонним.

# Задача (продолжение)

Задание:

Напишите набор тестов (специальные последовательности данных), которые будут адекватно проверять эту программу. Построив тесты, проанализируйте их.

## 2. Вопросы

Оцените Ваш набор тестов, попытавшись с его помощью ответить на приведенные ниже вопросы.

За каждый ответ "ДА" присуждается одно очко.

1. Составили ли Вы тест, который представляет правильный неравносторонний треугольник?  
(Ответ "ДА" на тесты со значением  $(1,2,3)$ ,  $(2,5,10)$  не обоснован, т.к. не существует треугольник, имеющий такие стороны).
2. Составили ли Вы тест, который представляет правильный равносторонний треугольник?

3. Составили ли Вы тест, который представляет правильный равнобедренный треугольник?  
(Значения  $(2,2,4)$  не засчитывать).
4. Составили ли Вы по крайней мере три теста, которые представляют правильные равнобедренные треугольники, полученные как перестановки двух равных сторон треугольника  
(например,  $(3,3,4), (3,4,3), (4,3,3)$ )?
5. Составили ли Вы тест, в котором длина одной из сторон треугольника принимает нулевое значение?
6. Составили ли Вы тест, в котором длина одной из сторон треугольника принимает отрицательное значение?

7. Составили ли Вы тест, включающий три положительных целых числа, сумма двух из которых равна третьей (например  $(1,2,3)$ )?
8. Составили ли Вы по крайней мере три теста с заданными значениями всех трех перестановок, в которых длина одной стороны равна сумме длин двух других сторон ( $(1,2,3), (1,3,2), (3,1,2)$ )?
9. Составили ли Вы тест из трех целых положительных чисел, таких что сумма двух из них меньше третьего числа ( $(1,2,4), (12,15,30)$ )?
10. Составили ли Вы по крайней мере три теста из категории 9, в которых Вами испытаны все три перестановки (например,  $(1,2,4), (1,4,2), (4,1,2)$ )?

11. Составили ли Вы тест, в котором все стороны треугольника имеют длину, равную нулю (0,0,0)?
12. Составили ли Вы по крайней мере один тест, содержащий нецелые значения?
13. Составили ли Вы по крайней мере один тест, содержащий неправильное число значений (например, два, а не три целых числа)?
14. Составили ли Вы по крайней мере один тест, содержащий нечисловые значения (например (A,2,3))?
15. Специфицировали ли Вы в каждом тесте не только входные значения, но и выходные данные программы?

Вопросы 1-13 представляют ошибки, имевшие место в различных версиях данной программы, адекватный тест для нее должен их обнаруживать.

Опытные профессиональные программисты набирают в среднем только 7-8 очков.

Задача и вопросы взяты из книги: Г.Майерс, "Искусство тестирования программ", М, Финансы и статистика, 1982

### 3. Документ определения требований (ДОТ) к задаче

- Функциональные требования:
  - 1. Чтение данных
  - 2. Интерпретация данных
  - 3. Вывод сообщения.

- Входные и выходные данные:
4. Вход - три целых положительных числа, которые интерпретируются как длины сторон треугольника.
- 5. Выход - сообщение:
  - треугольник - неравносторонний, равнобедренный или равносторонний;
  - сообщение об ошибке.
- 
- Нефункциональные требования:

# 4. Спецификация требований (СТ) к задаче

Функциональные требования:

- 1.1. Программа должна последовательно читать элементы набора данных, разделенные определенным разделителем.
- 1.2. Должно быть управление выбором вида разделителя между элементами набора данных (. , ; пробел).
- 1.3. Наборы должны отделяться друг от друга определенным образом.
- 1.4. Если обнаруживается ошибка при чтении данных, анализ данных не производится, выводится сообщение об ошибке.

- 2.1. Программа должна анализировать полученные данные и выводить соответствующее сообщение.
- 2.2. Если обнаруживается ошибка при анализе данных, выводится сообщение об ошибке.
- 3.1. По результатам анализа выводится одно или несколько сообщений.

## Входные и выходные данные:

- 4.1. Элемент набора - число
- 4.2. Элемент набора - целое число.
- 4.3. Элемент набора - целое положительное число.
- 4.4. Определены минимальное и максимальное значение элемента набора - 1 и 9999.
- 4.5. В наборе три элемента, разделенные разделителем.
- 4.6. Элементы набора должны интерпретироваться как длины сторон треугольника.

- 5.1. Сообщение 01 - неравносторонний треугольник, если ни одна из сторон не равна другой.
- 5.2. Сообщение 02 - равнобедренный треугольник, если две из сторон равны друг другу.
- 5.3. Сообщение 03 - равносторонний треугольник, если все три стороны равны друг другу. В данном случае сообщение о равнобедренном треугольнике не выводится.
- 5.4. Сообщение 04 - неправильный разделитель элементов набора данных
- 5.5. Сообщение 05 - неправильный разделитель наборов данных

- 5.6. Сообщение 06 - неправильный тип элемента набора данных
- 5.7. Сообщение 07 - выход за определенные границы данных
- 5.8. Сообщение 08 - неправильное количество элементов в наборе данных.
- 5.9. Сообщение 09 - входные данные не могут интерпретироваться как длины сторон треугольника.
- 5.10. Если при чтении или анализе входных данных обнаруживается ошибка, то выводится сообщение об ошибке (ошибках), сообщения 01-03 - не выводятся.

# 5. Наборы тестовых данных для задачи

## **Начальные условия:**

- разделитель между элементами набора данных - (,) - запятая
- разделитель между наборами данных - каждый набор данных на отдельной строке, ограничен скобками ( )
- диапазон данных 1...9999.

## **Чистый и грязный тест:**

Чистый тест (позитивный тест) - тест с правильным набором данных. Проверка соответствия.

Цель - демонстрация корректной работы программы.

В одном тесте данные могут соответствовать многим требованиям.

Все чистые тесты могут быть сведены к небольшому количеству.

Грязный тест (негативный тест) - тест с неправильным набором данных. Проверка искаженности.

Цель - проверка обработки программой искаженных (неправильных) данных.

Цель - взломать программу.

В одном тесте искажается только одно требование.

Грязные тесты желательно проверять все.

При формировании входных наборов данных указываются также ожидаемые результаты и сообщения.

1.1.1. Правильный разделитель в наборе данных

(2,3,4) - 01 (Сообщение)

1.1.2. Неправильный разделитель в наборе данных

(2.3,4) - 04

(2,3.4) - 04

1.2. - нет (пока) наборов данных для проверки возможности смены разделителя между элементами.

1.3.1. Правильный ограничитель набора данных

(3,3,4) - 02

1.3.2. Неправильный ограничитель набора данных

[2,3,4] - 05

- Типы данных:
  - буквы (литеры)
  - цифры (числа)
  - группы специальных символов
  - прочие символы

4.1.1. Элемент набора - число  
(3,3,3) - 03

4.1.2. Элемент набора - не число, по очереди на каждом месте, а также во всех местах

(А,3,3) - 06 - Литера - 1 позиция

(3,А,3) - 06

(3,3,А) - 06

(А,А,3) - 06 - 2 позиции

(А,3,А) - 06

(3,А,А) - 06

(А,А,А) - 06 - 3 позиции

(-,3,3) - 06 - Символ - 1 позиция

(3,!,3) - 06

(3,3,%) - 06

(-,!,3) - 06 - 2 позиции

(-,3,?) - 06

(3,!,?) - 06

(!,/,?) - 06 - 3 позиции

4.2.1. Элемент набора - целое число

(4,5,6) - 01

4.2.2. Элемент набора - не целое число, по очереди на каждом месте, а также во всех местах

$(-2.5, 5, 6) - 06$

$(5, 0.5, 6) - 06$

$(5, 5, 6.6) - 06$

$(-2.5, +5.5, 0.6) - 06$

4.3.1. Элемент набора - целое положительное число

$(5, 5, 6) - 02$

### 4.3.2. Элемент набора - не целое положительное число, по очереди на каждом месте, а также во всех местах

$(0,5,6)$  - 06 - Ноль как элемент набора - 1  
позиция

$(5,0,6)$  - 06

$(5,5,0)$  - 06

$(0,0,5)$  - 06 - 2 позиции

$(0,5,0)$  - 06

$(5,0,0)$  - 06

$(0,0,0)$  - 06 - 3 позиции

$(-2, 5, 6)$  - 06 - Отрицательное число как элемент набора

$(5, -2, 6)$  - 06

$(5, 5, -2)$  - 06

$(-4, -4, 6)$  - 06 - 2 позиции

$(-4, 5, -4)$  - 06

$(5, -4, -4)$  - 06

$(-2, -2, -2)$  - 06 - 3 позиции

## 4.4.1. Проверка минимальной и максимальной границы наборов данных

(1,2,2) - 02 - Проверка минимальной границы

(2,1,2) - 02

(2,2,1) - 02

(1,1,1) - 03

(9999,5001,5002) - 01 - Проверка максимальной границы

(5001,9999,5002) - 01

(5001,5002,9999) - 01

(9999,9999,9999) - 03

## 4.4.2. Выход за границы интервала.

Выход за минимальные границы - проверен в п. 4.3.2.

(10000,5005,5005) - 07 - Проверка выхода за  
максимальные границы

(5005,10000,5005) - 07

(5005,5005,10000) - 07

(10000,10000,10000) - 07

4.5.1. В наборе 3 элемента

(4,4,4) - 03

## 4.5.2. В наборе не 3 элемента

(,,) - 08 - 0 элементов

(1,,) - 08 - 1 элемент

(1,2,) - 08 - 2 элемента

(1,1,1,1) - 08 - 4 элемента

## 4.6.1. Входные данные могут интерпретироваться как длины сторон треугольника

(2,3,4) - 01 -  $a+b>c$

(2,3,2) - 02 -  $a+c>b$

(4,4,4) - 03 -  $b+c>a$

## 4.6.2. Входные данные не могут интерпретироваться как длины сторон треугольника

$$(1,2,3) - 09 - a+b=c$$

$$(1,3,2) - 09 - a+c=b$$

$$(3,1,2) - 09 - b+c=a$$

$$(1,2,4) - 09 - a+b<c$$

$$(1,4,2) - 09 - a+c<b$$

$$(4,1,2) - 09 - b+c<a$$