

Разбор задачи “Пифагоровы тройки” Чемпионата
Республики Карелия по командному
программированию среди школьников.
Октябрь 2010.

Итак, напомним коротко условие задачи. Необходимо было вывести k -е по счету алгебраическое уравнение $c^2 = a^2 + b^2$ в Васином списке. Напомним, что у Васи число a было всегда меньше, либо равно, числа b . В списке, уравнения были расположены таким образом, что если у одной пифагоровой тройки число c было меньше числа c другой пифагоровой тройки, то первая тройка располагалась раньше. Если же число c первой тройки совпадало с числом c второй тройки, то сравнивались числа a по аналогии. А если же числа a тоже совпадали, то сравнивались числа b .

Из условия задачи сразу же следует наивная реализация поставленной задачи:

1. Счетчик количества пифагоровых троек *count* сделать равным 0.
2. Перебирать число c от 1, пока не будет найдено k -е уравнение.
3. Для каждого c перебирать число a от 1 до c .
4. Для каждого a перебирать число b от a до c .
5. Если $c^2 = a^2 + b^2$, то $count = count + 1$.
6. Если $count = k$, выводим c , a , b и выходим из программы.

Можно заметить, что сложность этого алгоритма $O(c^3)$. При максимальном ограничении c , равном 2000, программа выполнит порядка $2000^3 = 8000000000 = 8 \cdot 10^9$ итераций возведения в квадрат и сравнения. Выполнение данного количества итераций, к сожалению, не укладывается в отведенное ограничение по времени (современные компьютеры выполняют порядка $10^8 - 10^9$ простейших итераций в секунду. Итерации сравнения и возведения в квадрат не являются простейшими).

Но!, можно заметить, что выполнение шага 4 совершенно напрасно. Действительно, на данном шаге нам всего-лишь нужно решить уравнение вида $b = \sqrt{c^2 - a^2}$. И если число b оказывается целым, и при том не меньше a , то можно сразу выполнить $count = count + 1$ и перейти к шагу 6.

Таким образом, при максимальном ограничении s , равном 2000, программа выполнит порядка $2000^2 = 4000000 = 4 \cdot 10^6$ итераций извлечения квадратного корня и сравнения. Выполнение данного количества итераций прекрасно укладывается в отведенное ограничение по времени.

Более подробно детали реализации можно посмотреть в решениях жюри.

Разбор подготовил Басунков Владимир Андреевич, преподаватель Клуба творчества программистов ПетрГУ.