BabylonianAlgorithm

Your task is to write a program that will find the root of a user specified value by using the Babylonian Algorithm.

EXAMPLE 4.11 The Babylonian Algorithm for Computing Square Roots

Over 5000 years ago the ancient Babylonians discovered a method for computing the square root of 2. They probably used that number (about 1.4) to construct right angles for the foundations of their buildings. This iterative algorithm is still the simplest way to compute square roots.

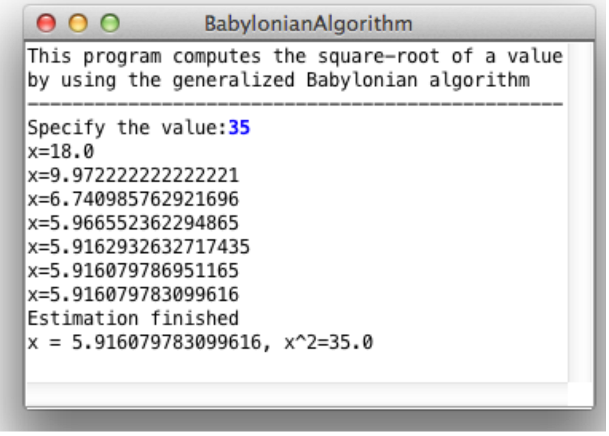
If x is any number close to $\sqrt{2}$, then x^2 will be close to 2, which makes x close to $2/x$. But $2/x$ will be on the other side of $\sqrt{2}$ from x. That is, if x less than $\sqrt{2}$, then $2/x$ will be greater than $\sqrt{2}$, and *vice versa*. For example, suppose that $x = 1.6$. Then $2/x = 1.25$, which is on the other side of $\sqrt{2}$. That "crossing over the limit" is the key to the algorithm because it means that the average of x and $2/x$ must be between them and therefore closer to the objective $\sqrt{2}$. So the Babylonian Algorithm consists of choosing some number x that is close to $\sqrt{2}$, and then repeatedly replacing x by its average with $2/x$. That's what the statement

```
x = (x + 2.0/x)/2;
```

The above definition indicates the solution for finding the square root of 2. However, the approach can be generalized to find the square-root of any positive real value, ***c***, using the following iterative solution:

$$x = (x + \frac{c}{x})/2$$

Here is a sample output for your program:



```
This program computes the square-root of a value
by using the generalized Babylonian algorithm
------------------------------------------------
Specify the value:35
x=18.0
x=9.972222222222221
x=6.740985762921696
x=5.966552362294865
x=5.9162932632717435
x=5.916079786951165
x=5.916079783099616
Estimation finished
x = 5.916079783099616, x^2=35.0
```

Resource:
http://www.mhprofessional.com/product.php?isbn=0071420401
Schaum's Outline of Programming with Java, by John Hubbard, McGraw Hill