Self Test: While and for loops

1. A band of 17 pirates stole a sack of gold coins. When they tried to divide the fortune into equal portions, three coins remained. In the ensuing brawl, one pirate was killed. The wealth was redistributed, but this time, 10 coins remained. Again, an argument arose during which one more pirate was killed. But now, the fortune was evenly distributed among the survivors. What is the smallest possible number of coins?
   [Hint: The number of coins divided by 17 gives remainder 3, divided by 16 remainder 10 and divided by 15 has remainder 0. Just try out all possible values for the number of coins starting with 1.] Solution is pirates.py.

2. Heron's method calculates an approximation to the square-root. If S is the number whose square-root is to be taken, and a is an approximation of the square-root, then Heron's method calculates a better approximation by

$$a_{\text{better}} = \frac{1}{2}\left(a + \frac{S}{a}\right)$$

   For example, if $S = 2$ and we choose $a = 1$, then the better approximation according to Heron's is 1.5. If we apply the method again, then we get 1.4166666666666665. Another application gives us 1.4142156862745097, which is already accurate to the fifth digit after the decimal. Write a program that asks the user for a number. Calculate an approximation of the square-root by starting out with initial approximation 1 and then running Heron's method twenty times to get a better approximation. (Hint: You need to use a for loop. In the body of the for-loop, you improve the value of a.)