8 Instruction 2IP65 and 2IP70

8.1 Collatz

A *Collatz sequence* is a sequence of integer numbers $a_0, a_1, a_2, \ldots$ that is obtained as follows.

1. start with an arbitrary positive number a_0;

2. if a_n is even, then $a_{n+1} = a_n/2$;

3. if a_n is odd and $a_n > 1$, then $a_{n+1} = 3a_n + 1$;

4. if $a_n = 1$, the sequence ends with a_n.

Example: 12, 6, 3, 10, 5, 16, 8, 4, 2, 1.

In 1937, the German mathematician Lothar Collatz proposed the conjecture (*vermoeden*) that these sequences are all finite. Since then, it has been shown that for many starting numbers the sequence ends, but no proof of the conjecture has been found yet. Other names are *Ulam conjecture* and *Syracuse problem*. The sequence is sometimes called the *hailstone sequence* (from the movement of hailstones while they are formed, I guess).

1. Write a program that reads from input a positive number and then prints the Collatz sequence starting with that number, followed by the length of the sequence and the maximum number reached.

2. Make this into a game by asking two starting numbers and then printing side by side the sequences. Use a tab character to separate them (`"\t"`). Example:

   ```
   16 8
   8 4
   4 2
   2 1
   1
   Player 1 wins with 5 to 4.
   ```

3. Make the game more interesting by specifying an interval from which the starting values has to be chosen. The interval is specified with lower and upper bound (maybe it's fair to have one player enter the lower bound and the other one the upper bound). When a player enters a number outside the interval, he has to repeat his entry.

4. Forbid entering values that were previously used. So remember all the starting numbers in an `ArrayList<Integer>`. Search through the ArrayList to see whether the number has been entered already. You may use the methods *contains* or *indexOf* of ArrayList.

5. Try to find a better winning criterion than the length of the sequence. Is the length divided by the starting number a good criterion, or does it favour low numbers too much? Maybe dividing by the log of the starting number is a good one?

6. * If you keep the ArrayList sorted, the search method can be more efficient. Implement this.

Kees Huizing – 19 October 2009