Absolute Oddness

by Sven Nilsen, 2016

When searching for symmetric paths, I discovered something curious. The formulas I used for even and odd numbers were the following:

even(x) = (x%2)==0
odd(x) = (x%2)==1

Usually we think about even and odd numbers as two mirror images. If a whole number does not belong to one group, it belongs to the other.

Yet, on closer inspection this turns out to be false. It is true for *positive whole numbers*, but false for negative.

Because I made the assumption that it was true for all whole numbers, I could not understand at first why the search did not connect `add[odd] <=> eq`, but found `add[even] <=> eq` only.

The reason was that negative numbers were included in the search input! If you take modulus 2 (%2) of a number, you get one element of the set {0, 1, -1}.

In order to fully make sense of odd numbers, you need 3 different functions:

odd(x) = (x%2)==1
neg_odd(x) = (x%2)==-1
abs_odd(x) = abs(x%2)==1

When using both positive and negative numbers, then `abs_odd` is the only mirror image of `even`.

For positive numbers:

abs_odd <=> odd

This means that if you search with positive numbers, then for each symmetric path that uses `odd` there is a symmetric path using `abs_odd`.