Towers of Hanoi

Björn Formgren

Spring 2023

Introduction

The fourth assignment in ID1019 consisted of solving the tower of Hanoi puzzle in Elixir. This is mainly an assignment in recursion since the code itself is not lengthy.

Hanoi

The puzzle consists of three pegs and a number n discs. The goal is to move all discs from the first peg to the last peg while adhering to a set of rules.

- Larger discs can only be places on smaller discs.
- Only one disc at a time can be moved.
- Only the upmost disc in a stack can be moved.

In the towers of Hanoi puzzle every solution requires $M(n) = 2*M(n-1)+1$ moves, this can be simplified mathematically to $2^n - 1$. The base case came from moving 0 discs where the number of moves $M(0) = 0$. Every solution of n discs is depending on the former iteration of moving $n-1$ discs. This means that if we know how to solve the puzzle for zero disc, we know how to do it for one disc and for two and so on.

```
def hanoi(0,_,_,_) do [] end
```

The recursion consisted of three parts. For an iteration of n discs we first want to move the $n-1$ discs to the aux peg using the third peg as the aux peg. Secondly we want to move the largest disc to it's destination peg. Lastly we want to the move the $n-1$ discs on top of the largest disc.

```
def hanoi(n, a, b, c) do
    hanoi(n-1, a, c, b) ++ [{:move, a, c}] ++ hanoi(n-1, b, a, c)
  end
```

If we call the hanoi function with:

```
    hanoi(3, :a, :b, :c)
```

The process goes as follows. The first recursive call will solve the tower for $n-1$ discs which are 2 in this case. The concatenating of the move a to c represents moving the largest disc to it's destination. The last recursive call represents moving the $n-1$ discs on top of the largest disc.

For a tower of Hanoi puzzle with four discs the sequence looks like this:

```
iex(19)> Hanoi.hanoi(4, :a, :b, :c)
[
  {:move, :a, :b},
  {:move, :a, :c},
  {:move, :b, :c},
  {:move, :a, :b},
  {:move, :c, :a},
  {:move, :c, :b},
  {:move, :a, :b},
  {:move, :a, :c},
  {:move, :b, :c},
  {:move, :b, :a},
  {:move, :c, :a},
  {:move, :b, :c},
  {:move, :a, :b},
  {:move, :a, :c},
  {:move, :b, :c}
]
```

The number of moves required for four discs are $2^4 - 1$ which equals to 15. By doing the same calculation we can solve the number of moves for a tower of 10 discs. $2^{10} - 1 = 1023$ moves.