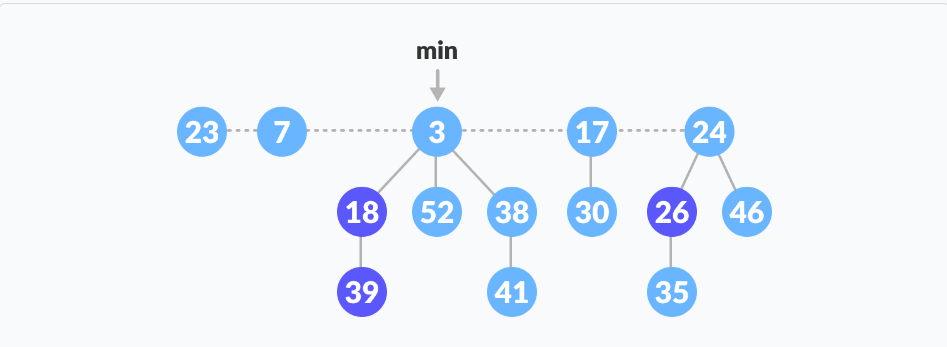
**Project Proposal**

I want to implement a Fibonacci heap. A Fibonacci heap is a collection of trees which follow the min or max heap property. I will implement a min Fibonacci heap. Nodes in the heap can have any number of subnodes, or zero subnodes. This data structure is called a Fibonacci heap because the trees are constructed such that any tree of order n has at least Fn+2 nodes in it (Fn+2 being the n+2­th Fibonacci number). 

*Picture credit:* [*https://www.programiz.com/dsa/fibonacci-heap*](https://www.programiz.com/dsa/fibonacci-heap)

I chose this data structure because it is efficient, and it has real-world use. The worst-case time is O(log n) for the delete min operation. The running time for other operations is Θ(1). This data structure can be used to improve the running time of algorithms such as Dijkstra’s shortest path algorithm.

The operations I will implement are:

1. findMin - Θ(1)
2. deleteMin O(log n)
3. insert - Θ(1)
4. decreaseKey (used in Dijkstra’s algo) - Θ(1)
5. merge (merge two fib heaps) - Θ(1)