

Questions

1. First I would figure out when do I want the recursive method to stop. If I want the method to stop at $f(n) = 0$, I would write an if statement when $f(n) = 0$, then return something. If I want the method to stop at $f(n) = 2$, I would do the same thing.

2. Pros: Recursion provides a more intuitive solution.

Cons: Recursion can be expensive in terms of processor time, it is exponential time, which can grows very quickly. Sometimes, iteration can provide a more efficient solution, it uses polynomial time.

3. A Collection can be order or unordered. A List contains ordered collection of objects known as sequence. List allow access by index.

To check if an object is List without looking at the source code, you can use "instanceof" keyword.

4. An Iterator is an object use to traverse the elements stored in the data structure. Iterator keeps track of current iteration and knows what next element is. Methods of an Iterator include hasNext(), next(), and remove(). Implementing Interface Iterable allows an object to be the target of the "foreach" statement. Iterable contains only one method Iterator<T> iterator().

Use Iterator when you want to check if there is a next element, to retrieve the next element, or to remove an element.

5. True, because one edge connects two nodes, and if you add one more node, you add one more edge. 3 nodes, $3-1 = 2$ edges.

6. 1,2,3,4,5,6,9,12

References

Fung, Carol. "Linked List." Lecture.

Goodrich, Michael T., and Roberto Tamassia. Data Structures and Algorithms in Java. New York: John Wiley, 2014. Print.