Recursion

Recursion is a very powerful concept in Computer Science. It is a way of defining something in terms of itself. A recursive solution to a problem is obtained through solving simpler versions of the original problem while a recursive description of a structure is obtained through the description of simpler versions of the original structure.

A recursive program is one that calls itself. A recursive program cannot call itself always, or it would never stop, so a second essential ingredient is that there must be a termination condition when the program can cease to call itself.

Recursion in Mathematics

Here is a simple geometric sequence:
>   3, 6, 12, 24, …

If we call the terms t_1, t_2, t_3,… then we can express the relationship concisely as follows:
> $t_1 = 3$
> $t_n = 2 \times t_{n-1}$, if $n > 1$

This definition of the terms of the sequence satisfies the conditions that we set out for any recursive definitions:
1. An object is defined in terms of another objects of the same type. Here a term is defined in terms of preceding term.
2. There is some way of stopping the recursion. This is done by explicitly defining the first term, t_1, as having the value 3.

This definition can be easily translated in to a recursive program in Java.

```java
public static int t(int n) {
   if (n == 1) {
      return 3;
   } else {
      return 2 * t(n-1);
   }
}
```