Induction and Recursion
Recursive Definitions and Structural Recursion

March 28, 2016
Recursive Definition

Recursive Functions

Definition
A function defined in terms of values of the same function on smaller (simpler) values derived from the parameters is a recursively-defined function. In addition to a recursive step, there must be a basis step (a base case).

Example

\[ \begin{array}{c|c}
  k & f(k) \\
  \hline
  0 & 3 \\
  1 & 4 \\
\end{array} \]

\[
\begin{align*}
f(0) &= 3 \\
f(n + 1) &= 2f(n) + 3
\end{align*}
\]
Recursive Definition

Recursive Functions

Definition

A function defined in terms of values of the same function on smaller (simpler) values derived from the parameters is a recursively-defined function. In addition to a recursive step, there must be a basis step (a base case).

Example

\[
\begin{array}{cc}
k & f(k) \\
0 & 3 \\
1 & 9 \\
2 &
\end{array}
\]

\[
f(0) = 3 \\
f(n + 1) = 2f(n) + 3
\]
Recursive Definition

Recursive Functions

Definition
A function defined in terms of values of the same function on smaller (simpler) values derived from the parameters is a recursively-defined function. In addition to a recursive step, there must be a basis step (a base case).

Example

\[
\begin{array}{ccc}
  k & f(k) \\
  0 & 3 \\
  1 & 9 \\
  2 & 21 \\
  3 &
\end{array}
\]

\[
f(0) = 3 \\
f(n + 1) = 2f(n) + 3
\]
Recursive Definition

Recursive Functions

Definition

A function defined in terms of values of the same function on smaller (simpler) values derived from the parameters is a recursively-defined function. In addition to a recursive step, there must be a basis step (a base case).

Example

\[
\begin{array}{c|c}
 k & f(k) \\
 0 & 3 \\
 1 & 9 \\
 2 & 21 \\
 3 & 45 \\
 4 & \\
\end{array}
\]

\[
\begin{align*}
 f(0) & = 3 \\
 f(n + 1) & = 2f(n) + 3
\end{align*}
\]
Recursive Definition

Recursive Functions

Definition
A function defined in terms of values of the same function on smaller (simpler) values derived from the parameters is a recursively-defined function. In addition to a recursive step, there must be a basis step (a base case).

Example

\[
\begin{align*}
  f(0) &= 3 \\
  f(n + 1) &= 2f(n) + 3
\end{align*}
\]
Recursive Definition

Recursive Functions

Definition

A function defined in terms of values of the *same* function on smaller (simpler) values derived from the parameters is a *recursively-defined function*. In addition to a recursive step, there must be a basis step (a base case).

Example

\[
\begin{align*}
f(0) &= 3 \\
f(n + 1) &= 2f(n) + 3
\end{align*}
\]

<table>
<thead>
<tr>
<th>(k)</th>
<th>(f(k))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td>45</td>
</tr>
<tr>
<td>4</td>
<td>93</td>
</tr>
<tr>
<td>5</td>
<td>189</td>
</tr>
</tbody>
</table>
Recursive Definition

Examples

Example

Give a recursive definition of the power function: \( x^{(n+1)} \) defined in terms of lower powers of \( x \).
Recursive Definition

Examples

Example

Give a *recursive* definition of the power function: $x^{(n+1)}$ defined in terms of lower powers of $x$.

Example

Give a *recursive* definition of the factorial function.
Example

Give a *recursive* definition of the power function: $x^{(n+1)}$ defined in terms of lower powers of $x$.

Example

Give a *recursive* definition of the factorial function.

Example

Give a *recursive* definition of the $\sum_{i=0}^{n} a_i$.
Recursive Definition

Fibonacci Numbers

Example

The Fibonacci numbers, $f_0, f_1, f_2, \ldots$, are defined as

\begin{align*}
  f_0 &= 0, \quad f_1 = 1 \\
  f_n &= f_{n-1} + f_{n-2}
\end{align*}
Recursive Definition

Structural Recursion

Example

```c
int fib(int n) {
    if (n == 0) return 0;
    if (n == 1) return 1;
    return fib(n - 2) + fib(n - 1);
}
```
Recursive Definition

Structural Recursion

Let $\Sigma$ be an alphabet. Then let $\Sigma^*$ be the set of strings over $\Sigma$.
Example: $\Sigma = \{a, b\}$.
Then $\Sigma^* = \{\text{""}, \text{"a"}, \text{"b"}, \text{"aa"}, \text{"ba"}, \text{"ab"}, \text{"bb"}, \text{"aaa"}, \text{"baa"}, \text{"aba"}, \text{"bba"}, \ldots\}$.

Recursive Definition of $\Sigma^*$:
Basis Step: "" $\in \Sigma^*$
Recursive Step: If $w \in \Sigma^*$ and $x \in \Sigma$ then $wx \in \Sigma^*$. 
Definition

length of a string.

Basis Step: The length of "" is 0. Write this as \( l("") = 0 \)

Recursive Step: If \( w \in \Sigma^* \) and \( x \in \Sigma \) then \( l(wx) = l(w) + 1 \).
Recursive Definition

A rooted tree.

Basis Step: A single vertex is a rooted tree.

Recursive Step: Suppose that $T_1, T_2, \ldots, T_n$ are disjoint rooted trees with roots $r_1, r_2, \ldots, r_n$ respectively. Then the graph formed by taking vertex $r$ (where $r \neq r_i, 1 \leq i \leq n$) and adding an edge from $r$ to each of $r_1, r_2, \ldots, r_n$ is a rooted tree.

Example

Give a recursive definition of the $height$ of a rooted tree.
Definition

Full Binary Tree

Basis Step: A single vertex \( r \) is a full binary tree.  
Recursive Step: Suppose that \( T_1 \) and \( T_2 \) are disjoint full binary trees with roots \( r_1 \) and \( r_2 \) respectively. Then for a new vertex \( r \) (\( r \) is not a vertex in either \( T_1 \) or \( T_2 \)), \( r \) together with edges to \( r_1 \) and \( r_2 \) is the root of binary tree \( T \).
Recursive Definition

Structural Recursion

Example

Height of a full binary tree.
Basis Step: The full binary tree $T$ consisting of a single root $r$ has height 0, that is $h(T) = 0$.

Recursive Step: If $T_1$ and $T_2$ are full binary trees, and the immediate subtrees of full binary tree $T$, then $h(T) = 1 + \max(h(T_1), h(T_2))$. 

Recursive Definition

Structural Recursion

Example

Number of vertices in a full binary tree - $n(T)$.

Basis Step: The full binary tree $T$ consisting of a single root $r$ has 1 vertex, that is $n(T) = 1$.

Recursive Step: If $T_1$ and $T_2$ are full binary trees and immediate subtrees of full binary tree $T$, then $n(T) = 1 + n(T_1) + n(T_2)$. 