Disc Math - § 8 - Recursive Definitions

Def: A definition of a sequence of objects (numbers, sets, logical formula functions, ets) is **recursive** if an object from the sequence is defined in terms of earlier members of the sequence (recursion over a directed set later).

Ex: $S_1 = 0$, $S_{n+1} = 2S_n + 1$ → $0, 1, 3, 7, 15, 31, \ldots$

Ex: (The Fibonacci sequence).

$F(1) = 1$, $F(2) = 1$, $F(n) = F(n-1) + F(n-2)$, $n > 2$ [Also $F(0)=0$].

→ $1, 1, 2, 3, 5, 8, 13, 21, 34, 55, \ldots$

Ex: Prove that $F(n+4) = 3F(n+2) - F(n)$, $n \geq 1$. Strong induction.

Base cases: $n=1$ $\quad 5 = F(5) = 3F(3) - F(1) = 3(2) - 1$ ✓

$n = 2$ $\quad 8 = F(6) = 3F(4) - F(2) = 3(3) - 1$ ✓

Assume $F(i+4) = 3F(i+2) - F(i)$, $1 \leq i \leq k$. Need to prove

$$F(k+5) = 3F(k+3) - F(k+1).$$

$$F(k+5) = F(k+4) + F(k+3) \underset{\text{ind. ass.}}{=} 3F(k+2) - F(k) + 3F(k+1) - F(k-1) =$$

$$= 3[F(k+2) + F(k+1)] - [F(k) + F(k-1)] = 3F(k+3) - F(k+1). \quad \boxtimes$$

Ex: $$F(n) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

Pr: $p = \frac{1+\sqrt{5}}{2}$, $q = \frac{1-\sqrt{5}}{2}$, $p+q=1$, $p^2 = 1+p$, $q^2 = 1+q$; $F(n) = \frac{p^n - q^n}{p-q}$

Base case: $n = 1$; $1 = F(1) = \frac{p-q}{p-q} = 1$ ✓

$n = 2$, $1 = F(2) = \frac{p^2 - q^2}{p-q} = p + q = 1$ ✓ $\quad$ Assume $F(i) = \frac{p^i - q^i}{p-q}$, $1 \leq i \leq k$

Need to prove: $F(k+1) = \frac{p^{k+1} - q^{k+1}}{p-q}$

$$\frac{p^{k+1} - q^{k+1}}{p-q} = \frac{p^2 p^{k-1} - q^2 q^{k-1}}{p-q} = \frac{(1+p)p^{k-1} - (1+q)q^{k-1}}{p-q} = \frac{p^k - q^k}{p-q} + \frac{p^{k-1} - q^{k-1}}{p-q} =$$

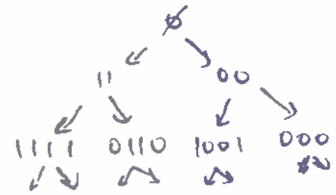
$= F(k) + F(k-1) = F(k+1)$ $\boxtimes$ $\quad$ rem: $\lim_{n\to\infty} \frac{F_{n+1}}{F_n} = p = \frac{1+\sqrt{5}}{2}$

Disc Math – §8 – Recursive Definitions

Ex: The formulas of propositional logic are built recursively:
- → any statement letter is a formula
- → if P, Q are formulas, so are $(P \wedge Q), (P \vee Q), (P \to Q), (P'), (P \leftrightarrow Q)$

e.g. $(P')'$, $P \to Q \vee R$ (sometimes we omit brackets).

Ex: A set S of strings is recursively defined by:



i) $\emptyset \in S$

ii) If $x \in S$, then $1x1 \in S$ and $0x0 \in S$.

We can use structural induction to prove that every string in S has even numbers of 0's and of 1's.

Pr: Base case $\emptyset$ has zero 0's, 1's → even. If $x \in S$ has even number of 0's and 1's so do $1x1$, $0x0$.

Ex: A recursive definition for the multiplication of two positive integers m, n is

① $m \cdot 1 = m$

② $m \cdot n = m \cdot (n-1) + m$, $n \geq 2$.

Recursive algorithms (recursively defined algorithms).

Ex: Consider a sequence $\{S_n\}_{n=1}^{\infty}$ defined recursively by:

① $S_1 = 2$

② $S_{n+1} = 2 S_n$, $n \geq 1$. It is: $2, 4, 8, 16, \ldots$

We can code this as a loop or as a recursive algorithm:

Loop: $S(n \in \mathbb{N})$ ← Defines the function $S(n)$.

(Iterative).

```
i - integer ; val
    if n = 1 then
        return 2
    else
        i = 2 , val = 2
        while i ≤ n do
            val = 2*val
            i = i + 1
        end while
        return val
    end if
```

Disc Math - §8 - Recursive Definitions

Recursive Algorithm:

```
S(n ∈ N)
    if n=1 then
        return 2
    else
        return 2 * S(n-1)
    end if.
```

rem: A recursive algorithm invokes itself with "smaller" input values.

ex: Goal: Sort a list L of n items into increasing or decreasing alphabetical / numerical order.

```
SelectionSort (list L, j ∈ N)          (starts with j=n).
    if j=1 then
        sort is complete, write the sorted list
    else
        find the index i of the max item in L between 1 and j
        exchange L[i] and L[j]
        selection sort (L, j-1)
    end if.
```

e.g.

Xeno	$\xrightarrow{SS(L,4)}$	Anik	$\xrightarrow{SS(L,3)}$	Anik	$\xrightarrow{SS(L,2)}$	Anik	$\xrightarrow{SS(L,1)}$	"Print"
Rita		Rita		Nita		Nita		
Nita		Nita		Rita		Rita		
Anik		Xeno		Xeno		Xeno		

HW §3.1 p 171

~~20~~, 25, 32, ~~52~~, 54, 74, ~~78~~, ~~81~~
~~82~~