02157 Functional programming

Michael R. Hansen
DTU COMPUTE
August 31, 2017

Exercises for Sept. 8^{th}

1. Declare a recursive function `f: int -> int`, where

$$\texttt{f}(n) = 1 + 2 + \cdots + (n-1) + n$$

for $n \geq 0$. (Hint: use two clauses with `0` and `n` as patterns.)
State the recursion formula corresponding to the declaration.
Give an evaluation for $\texttt{f}(4)$.

2. Declare a recursive function `sum: int * int -> int`, where

$$\texttt{sum}(m, n) = m + (m+1) + (m+2) + \cdots + (m + (n-1)) + (m+n)$$

for $m \geq 0$ and $n \geq 0$. (Hint: use two clauses with `(m,0)` and `(m,n)` as patterns.)
Give the recursion formula corresponding to the declaration.

3. The following figure gives the first part of Pascal's triangle:

```
        1
      1   1
    1   2   1
  1   3   3   1
1   4   6   4   1
```

The entries of the triangle are called *binomial* coefficients. The k'th binomial coefficient of the n'th row is denoted $\binom{n}{k}$, for $n \geq 0$ and $0 \leq k \leq n$. For example, $\binom{2}{1} = 2$ and $\binom{4}{2} = 6$. The first and last binomial coefficients, that is, $\binom{n}{0}$ and $\binom{n}{n}$, of row n are both 1. A binomial coefficient inside a row is the sum of the two binomial coefficients immediately above it. These properties can be expressed as follows:

$$\binom{n}{0} = \binom{n}{n} = 1$$

and

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k} \text{ if } n \neq 0,\ k \neq 0, \text{ and } n > k.$$

Declare an F# function `bin: int * int -> int` to compute binomial coefficients.

4. Declare an F# function `multiplicity`(x, ys) to find the number of times the value x occurs in the list ys. For example, `multiplicity(2, [2; 4; 2; 10; 1; 2]) = 3`.

5. Declare a function `mulC: int * int list -> int list` that multiplies every element of an integer list by a constant. For example, `mulC(2, [4; 10; 1]) = [8; 20; 2]`.

6. Declare a function `addE: int list * int list -> int list` that adds the elememts of two integer lists element by element. For example,

$$\begin{array}{lcl} \texttt{addE}([1;2;3],[4;5;6]) & = & [5;7;9] \\ \texttt{addE}([1;2],[3;4;5;6]) & = & [4;6;5;6] \\ \texttt{addE}([1;2;3;4],[5;6]) & = & [6;8;3;4] \end{array}$$

7. We represent the polynomial $a_0 + a_1 \cdot x + ... + a_n \cdot x^n$ with integer coefficients $a_0, a_1, ..., a_n$ by the list $[a_0; a_1; \ldots; a_n]$. For instance, the polynomial $2 + x^3$ is represented by the list $[2; 0; 0; 1]$.

   Note that the function `mulC` above implements multiplication of a polynomial by a constant. For example, $2 \cdot (2 + x^3) = 4 + 2x^3$ and $\texttt{mulC}(2, [2; 0; 0; 1]) = [4; 0; 0; 2]$.

   Furthermore, the function `addE` above implements addition af two polynomials. Compare, for example, the addition $(1 + 2x) + (3 + 4x + 5x^2 + 6x^3) = 4 + 6x + 5x^2 + 6x^3$ with the second example given above.

   (a) Declare a F# function `mulX` for multiplying a polynomial $Q(x)$ by x. For example, $x \cdot (2 + x^3) = 2x + x^4$. That is, $\texttt{mulX}[2; 0; 0; 1]$ should be $[0; 2; 0; 0; 1]$.

   (b) Declare a function `mul` for multiplication of polynomials in the chosen representation. The following properties are useful when defining the multiplication:

   $$\begin{array}{ll} 0 \cdot Q(x) & = 0 \\ (a_0 + a_1 \cdot x + ... + a_n \cdot x^n) \cdot Q(x) & \\ & = a_0 \cdot Q(x) + x \cdot \big((a_1 + a_2 \cdot x + ... + a_n \cdot x^{n-1}) \cdot Q(x)\big) \end{array}$$

   For example, $(2 + 3x + x^3) \cdot (1 + 2x + 3x^2) = 2 + 7x + 12x^2 + 10x^3 + 2x^4 + 3x^5$.

   (c) Declare a function to give a textual representation of a polynomial.