plab
LCP

Foundations of Computer Science
2001

Model Answer

Here is the solution for the representation by functions:

```
fun constpower c i = if i=0 then c else 0.0;

fun exppower i = if i=0 then 1.0 else exppower(i-1)/real i;

fun cmultpower (c,f) i : real = c * f i;

fun addpower (f,g) i : real = f i + g i;

fun multpower (f,g) i : real =
  let fun sum k = if k<=i then f k * g (i-k) + sum (k+1)
                  else 0.0
  in  sum 0  end;
```

Here is the solution for the representation by the datatype (lazy lists), which are covered in the course notes:

```
fun forever x = Cons(x, fn()=> forever x);

fun constpower c = Cons(c, fn() => forever 0.0);

fun exppower_aux (c,i) = Cons (c, fn () => exppower_aux (c/i, i+1.0));

val exppower = exppower_aux (1.0,1.0);

fun cmultpower (c, Cons(x,xq)) =
    Cons(c*x: real, fn () => cmultpower(c, xq()));

fun addpower (Cons(x,xq), Cons(y,yq)) =
    Cons(x+y: real, fn () => addpower(xq(), yq()));

fun multcoeff ([], [])      = 0.0
  | multcoeff (x::xs,y::ys) = x*y + multcoeff (xs,ys);

fun multpower_aux (Cons(x,xq), Cons(y,yq), xs, ys) =
    Cons(multcoeff(x::xs, rev (y::ys)),
         fn () => multpower_aux(xq(), yq(), x::xs, y::ys));

fun multpower (xq, yq) = multpower_aux (xq, yq, [], []);
```