Assignment 6 solution

See the video.

```
1. Unify(real -> t4,   t5 -> (string -> int))

     - Both arrow types, unify left/right sides
     - Unify(real, t5)
          - {t5 = real}
     - Unify(t4, string->int)
          - {t4 = string->int}
     - real -> string -> int

2. Unify(t1 -> (t2 -> real), int -> (t3 list -> real))

     - Both arrow types, unify left/right sides
     - Unify(t1, int)  {t1 = int}
     - Unify(t2->real, t3 list->real)
         - Both arrow types
         - Unify(t2, t3 list)  {t2 = t3 list}
         - Unify(real, real)  SUCCEEDS
     - int -> t3 list -> real

3. Unify((int list) list, t6 list)

     - Both application/ID types, both are list
     - Unify(int list, t6)  {t6 = int list}
     - int list list

4. Unify( (t7 list * t8) * t8,   (t9 * t10 list) * t9)

     - Both cross types, unify left/right
     - Unify(t7 list * t8,  t9 * t10 list)
     - Unify(t8, t9) {t8=t9}
     - Unify(t7 list * t9,  t9 * t10 list)
         - Both cross types, unify left/right
         - Unify(t7 list, t9)  {t9 = t7 list}
         - Unify(t9, t10 list)
         - Unify(t7 list, t10 list) {t7 = t10}

     - t10 list * t10 list * t10 list
```

```
fun crow xs =
  if null xs then 1
  else hd xs * crow (tl xs)

crow : t20 -> t21
xs : t20
null : t8 list -> bool
Unify(t20, t8 list) { t20 = t8 list }

crow : t8 list -> t21
xs : t8 list
Unify(t21, Int)

crow : t8 list -> int
hd: t9 list -> t9
hd xs: {t9=t8} t8
op* {t8=int}

crow : int list -> int

-------------------------------

fun down x = if x=0 then [] else x :: down (x-1)

down: t30 -> t31
x : t30
x=0 therefore {t30=int}
down: int -> t31
x : int
[] : t11 list {t31 = t11 list}

down: int -> t11 list
op:: t12 * t12 list -> t12 list
op::(x, down(x-1)) {t12 = int}
op::(x, down(x-1)) :: int list
Unify(t11 list, int list) {t11=in}

down: int -> int list

------------

fun edgar n = crow (down n)

edgar : t40 -> t41
n : t40
```

```
down : int -> int list
down n {t40=int}
down n : int list
crow : int list -> int
unify(int list, int list) succeeds
unify(t41, int) {t41 = int}

edgar : int -> int
```