

# Eliminating Variables and sequential flows

Rushikesh Joshi  
IIT Bombay

# Computing the length of a list

```
def length(l):  
    x = isnil(l)(0,1)  
    if (x==0): return (0)  
    else: return (1+length(tail(l)))
```

**eliminate variable x from the above implementation?**

# Computing the length of a list

- *variable  $x$  is absent*
- *if else statement is not required as `isnil()` function which returns a boolean is used directly*

```
def length(l):  
    x = isnil(l)(0,1)  
    if (x==0): return (0)  
    else: return (1+length(tail(l)))
```

```
def _length(l):  
    return (isnil(l)(0,(1+length(tail(l))))))
```

The Higher order function *map*  
*The function is using recursion  
already*

```
def map(f,l):  
  x=isnil(l)(0,1)  
  if (x==0): return (l)  
  h = f(head(l))  
  t = map (f, tail(l))  
  return (cons (h,t))
```

**Eliminate variables x, h, t?**

# The Higher order function *map* *all variables eliminated*

```
def map(f,l):  
  x=isnil(l)(0,1)  
  if (x==0): return (l)  
  h = f(head(l))  
  t = map (f, tail(l))  
  return (cons (h,t))
```

```
def _map(f,l):  
  return (isnil(l)(l,cons (f(head(l)),map(f,tail(l))))))
```

# Pushback an element in a list

```
def pushback(l,e):  
    x = (isnil(l)) (0,1)  
    if (x==0): return (cons(e,nillist()))  
    return (cons (head(l),pushback(tail(l),e)))
```

**Eliminate variable x?**

# Push back an element in a list

- Eliminate variable x

```
def pushback(l,e):  
    x = (isnil(l)) (0,1)  
    if (x==0): return (cons(e,nillist()))  
    return (cons (head(l),pushback(tail(l),e)))
```

**redrafted as follows:**

```
def push_back(l,e):  
    return (isnil(l)) (cons(e,nillist()),cons  
                (head(l),pushback(tail(l),e)))
```

# Append two lists

```
def append(l1,l2):  
    x = isnil(l2)(0,1)  
    if (x==0): return (l1)  
    h = head (l2)  
    l = push_back(l1,h)  
    return (append(l,tail(l2)))
```

**Eliminate variables x, h, l?**

# Append two lists

- variables x, h, l eliminated

```
def append(l1,l2):  
    x = isnil(l2)(0,1)  
    if (x==0): return (l1)  
    h = head (l2)  
    l = push_back(l1,h)  
    return (append(l,tail(l2)))
```

```
def _append(l1,l2):  
    return( isnil(l2) (l1,  
                    append( push_back(l1,head(l2)), tail(l2))))
```

# Use of variables in iteration

```
def find(e,l):  
    i = 0  
    while (i<length(l)):  
        if (e==head(l)):  
            return (true)  
        l = tail(l)  
        i=i+1  
    return (false)
```