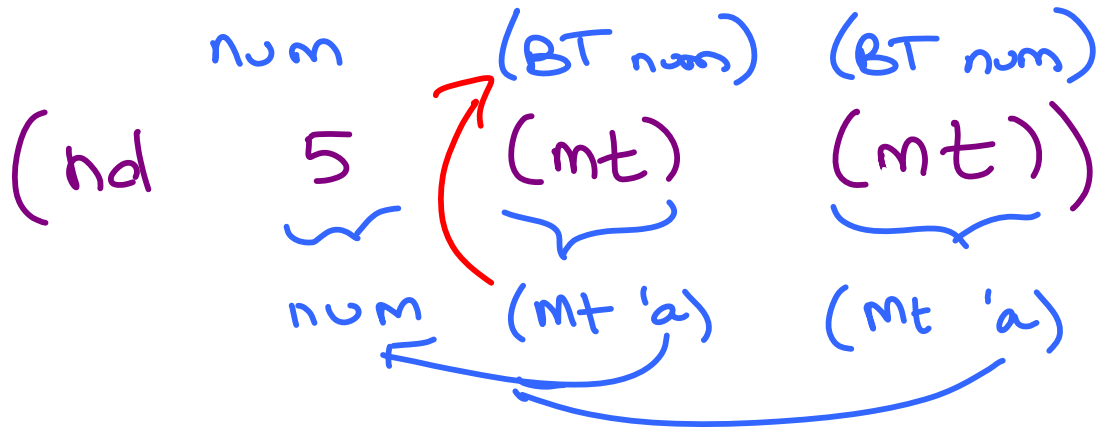
```
(d-t (BT 'a)
  [mt]
  [nd (v : 'a)
      (l : (BT 'a))
      (r : (BT 'a))])
```

```
mt : -> (BT 'a)

nd : 'a (BT 'a) (BT 'a)
     -> (BT 'a)

mt? : (BT 'a) -> Bool
nd? : (BT 'a) -> bool

nd-v : (BT 'a) -> 'a

nd-l : (BT 'a) -> (BT 'a)
nd-r
```

```
BT 'a    Mt 'a    Nd 'a

mt : -> Mt 'a
nd : 'a . BT 'a . BT 'a -> Nd 'a

nd-v : (Nd 'a) -> 'a
nd-l : (Nd 'a) -> (BT 'a)
nd-r
```

(BT 'a) ≡ (Nd 'a) ∪ (Mt 'a)



(Mt 'a)/(Nd 'a) can occur anywhere we expect a (BT 'a)

```
(define (size [t : (Bt 'a)]) : number
  (cond
    [(mt? t)   0]
    [(nd? t)   (+ 1
                  (size (nd-l t))
                  (size (nd-r t)))]))
```

```
(if (nd? t)
    (size (nd-l t))
    0)
```

$$\frac{\Gamma \vdash e_1 : \mathrm{Bool} \quad \Gamma \vdash e_2 : \tau \quad \Gamma \vdash e_3 : \tau}{\Gamma \vdash \mathrm{if}\ e_1\ e_2\ e_3 : \tau}$$

isinstance (o, c)

o.__class__

numbers

strings

objects

```
if (typeof x=='number'){
    ...
} else {
    ...
}
```

typeof 52 -> 'number'

Array<T> × Int × Int ∪ Undef

```
function slice(arr, start, stop) {
    if(typeof stop === 'undefined')
        stop = arr.length;
    var theslice = [];
    for(i=0; i<stop; i++){
        theslice[i] = arr[i];
    }
    return theslice;
}
```

Int ← stop = arr.length; → Undef

var theslice = []; → Int

slice([1,2,3,4], 1, 3)
→ [2,3]

slice([1,2,3,4], 2)
→ [3,4]

< : Int × Int → Bool

stop: Int ∪ Undef

stop : Int ∪ Undef