3EA3 Notes

Made by James Lee

Wednesday, March 15

Binary Search Refresher

$x = \uparrow i \cdot \mathbb{Z} \mid R\ i \qquad \equiv \qquad R\ x \wedge \neg\ R\ (x + 1)$

These two are equal when R is **sorted, finite, and non-empty**. Otherwise, binary search will just find an element with a neighbour that satisfies the property.

Provided $\mathcal{Z}$ is co-transitive
$\{a < b\}$

```
x, y := a, b
do x + 1 < y ->
    m := (x + y) / 2
    if m 𝒵 y -> x := m
    [] x 𝒵 m -> y := m  //[] is the guard symbol
    fi
od
```

$\{x \in a\ ..\ b - 1 \wedge x\ \mathcal{Z}\ (x + 1)\}$

Transitivity: $x\ \mathcal{Z}\ m \wedge m\ \mathcal{Z}\ y \Rightarrow x\ \mathcal{Z}\ y$
Co-transitivity: $x\ \mathcal{Z}\ m \vee m\ \mathcal{Z}\ y \Leftarrow x\ \mathcal{Z}\ y$
Duality: flip the signs ($\vee$ to $\wedge$ and vice versa) and reverse $\Rightarrow$ to $\Leftarrow$ and vice versa.

Example: Square

R: $a \leq x^2 \wedge a \leq x < b \wedge b < (x+1)^2$
Replace $a \leq x^2 \wedge b < (x+1)^2$ with $x\ \mathcal{Z}\ (x+1)$.

Tail Recursion - Just Loops!

```
L : do $B \rightarrow S$ od
```

is equal to

```
L: if $\neg B \rightarrow$ skip
    □ $B \rightarrow S$; L
    fi
```

Tail recursion has its recursion happen at the end!

```
H x = if $\neg B \rightarrow C\ x$
        □ $B \rightarrow H\ (d\ x)$
        fi
```

C x is the base case, and d x is performing an action ”delta” on x at each recursion level.
Two things to note: we need a bound function bf such that:
$\rightarrow 0 \leq bf$ ”start bf as non negative”
$\rightarrow bf[x:=dx] < bf$ ”make progress downwards”.

R (end goal): $r = H(N)$
P (invariant): $r = H\ n = H\ N$

```
n := N
{P: $r = H\ n$}
do $B \rightarrow n := d\ (n)$ od
{$\neg\ B \wedge H\ n = H\ N$}
r := C n
{$r = H(N)$}
```

Quick Factorial Example

```
fac n = if $n = 0 \rightarrow 1$
          □ $n \geq 1 \rightarrow n * fac(n-1)$
          fi
```