Recursion / fixpoint combinators

"combinator" $= \lambda x . x$

identity combinator

U combinator

$U = \lambda h .\ h\ h$

```
def fact(x):
    if x = 0 then return 1
    else x * fact(x-1)
```

Imagine we can do recursion in λ-calc

$fact = \lambda x.\ \text{if } (x = 0) \text{ then } 1 \text{ else } x * fact(x-1)$

$fact = \lambda h.\ \lambda x.\ \text{if } (x = 0) \text{ then } 1 \text{ else } x * (h\ h)(x-1)$

$fact\ fact$

$U(fact)$

fixedpoint combinators

$f : \mathbb{R} \to \mathbb{R}$

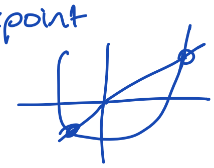
if $x \in \mathbb{R}$ s.t $x = f(x)$

then x is a fixed point of f

fixpoint

$f(x) = x^2 - 1$

$x = x^2 - 1$



$\lambda x . x$

$fact = \lambda x . . . .$

$F(f) = \lambda x.$ if $x = 0$ then 1 else $x * f(x-1)$

$\boxed{\text{find a fixed point of } F}$

Deriving the Y combinator

$F = \lambda x . . . . . . F . . .$

pull out the "unknown" F

$F = \underbrace{(\lambda f . \lambda x . . . . f . . . .)}_{h} F$

$(*)\quad F = h\,F \qquad Y\,h \longrightarrow f$

s.t. $f = h\,f$

---

let $F = F' F'$

therefore (following $*$) becomes

$F'\,F' = h\,(F'\,F')$

$= (\lambda x . h\,(x\,x))\,F'$

$\therefore\ F' = (\lambda x . h\,(x\,x))$

$x = (\lambda y . y)\,x$

$F = F'F' = (\lambda x . h\,(x\,x))\,(\lambda x . h\,(x\,x))$

$$= (\lambda h'.\ (\lambda x.\ h'(x x))\ (\lambda x.\ h'(x x)))\ h$$

$$Y$$

$$F = h\,F$$

$$F = Y\,h$$

$$F = Y\,h$$