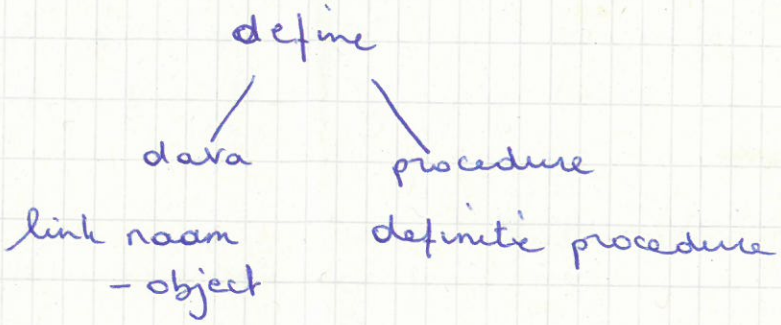
26/09/2019 expression ∈ combinatie



⇒ 2x zelfde define, telkens andere betekenis

function ≠ procedure

↳ always returns

↳ doesn't always return

onderscheid is niet in alle programmeertalen
vb. niet in Scheme

Conditional expressions Scheme:

- if / else
- if
- cond
- cond/else
  ↳ catch-all

Conditional ↔ iterative

vb. Newton approximation
geen iterative ???
⇒ recursion in functional programming in de plaats

check: 1quare defined in Scheme
≠ in Racket

check: Dr. Racket v6.8
language: R5RS
trace

Voordelen scoping:

1) portable (beter gepropeerd)
vb. deel sqr-root functie
⇒ deel 1 functie (die al helpers bevat)

voordelen scoping

2) geen collisie (encapsulatie)
   = interferentie
3) minder formele parameters nodig
   want al gedefinieerd in functie waar helper deel van is

lexical scoping
↳ waar variabelen bereikbaar zijn
↳ 'zoals het staat geschreven'

↕

dynamic scoping
runtime
gebruik variabele laatst tegengekomen

functie op 1 plaats gedefinieerd → duidelijk (LEXICAL)

variabelen kunnen op veel plaatsen gedefinieerd zijn → onduidelijk

gebonden ↔ vrije variabelen (DYNAMIC)

argumenten procedure

zoekt in omgeving waar procedure wordt opgeroepen