CSE216
Programming Abstraction

Instructor: Zhoulai Fu

State University of New York, Korea

Regarding quiz

- No more quiz: Seems it was more scary than supportive.
- First quiz results will be retained as-is.
- Overall grading formula remains the same

Plan

- Today: More on beta reduction

Sharing A good thing on lambda calculus

- I find this 3-page notes from Yale particularly clear:

- https://www.cs.yale.edu/homes/hudak/CS201S08/lambda.pdf

How beta reduction works

- Identify a redex $(\lambda\ x.M)N$
- Reduce the redex by substitution $M\ [x:=N]$
- Continue until redex found

Redex

- Reducible expression
- A redex is formed when a lambda definition is immediately followed by an argument, indicating that the function can be applied to that argument
- $(\lambda\ x.M)N$

Exercise: Identify redex

- $(\lambda\ x.\ \lambda\ y.\ x\ y)\ (\lambda\ s.\ s)$
- $(\lambda\ x.\ x)\ (\lambda\ y.\ y)$
- $(\lambda\ x.\ x\ x)\ (\lambda\ x.\ x\ x)$

Substitution

- (λ x.M)N -> M [x := N]
- For all occurrence of x in M:
    - If x is bound by the formal parameter “x” in (λ x.M)
    - the replace x by N in M
- We will consider capture-avoiding situations later

Normal form

- No redex

Exercise: Normal form or not

- λ x.λ y. xy
- λ x. x λ y.x y
- (λ x.λ y. x)y
- (λ x. x λ y.x)y

Exercise: beta reduction

- $(\lambda\ x.\ \lambda\ y.\ x\ y)\ (\lambda\ s.\ s)$
- $(\lambda\ x.\ x)\ (\lambda\ y.\ y)$
- $(\lambda\ x.\ x\ x)\ (\lambda\ x.\ x\ x)$

Does beta reduction order matter?

- Let D be the lambda term (λ x. xx). Reduce
  - (λ x.y)(D D)

Does beta reduction order matter?

- (λ x.y)**(**D D) -> (λ x.y)**(**D D) -> …
- (λ x.y)(D D) -> y
- The first one is call-by-name (lazy evaluation)
- The second one is call-by-value (eager evaluation)

Church-Rosser Theorem

- A lambda term can never have two different normal forms

Capture avoiding substitutions

- The specific names of bound variables in the lambda calculus are meaningless
- λ x. x same as λ y. y
- (λx.(λy.yx)) is equivalent to (λa.(λb.ba))
- Lambda terms that differ only by bound variable names are called alpha equivalence

Exercise: alpha equivalence?

- λ x. xy ? λ z. zy
- λ x. xy ? λ z. xz
- λ x. x λ y. y ? λ z. z λ p. p
- λ x. x λ y. y ? λ y. y λ y. y
- λ x. x λ y. x y ? λ y. y λ y. y y

Exercise: beta reduction

- $(\lambda z.z)\ (\lambda q.q\ q)\ (\lambda s.s\ a)$

Exercise: beta reduction

- (λz.z) (λz.z z) (λz.z q)

Exercise: beta reduction

- (λs.λq.s q q) (λa.a) b

Exercise: beta reduction

- $(\lambda s.\lambda q.s\ q\ q)\ (\lambda q.q)\ q$

Exercise: beta reduction

- $((\lambda s.s\ s)\ (\lambda q.q))\ (\lambda q.q)$

Solution

1. (λz.z) (λq.q q) (λs.s a) ->.. -> aa
2. (λz.z) (λz.z z) (λz.z q) -> .. -> qq
3. (λs.λq.s q q) (λa.a) b -> .. -> bb
4. (λs.λq.s q q) (λq.q) q-> .. -> qq
5. ((λs.s s) (λq.q)) (λq.q)-> .. -> λ q. q