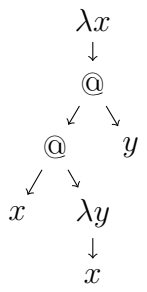
# CS 421 --- Lambda Calculus Activity

| | | |
|---|---|---|
| Manager | Keeps team on track | |
| Recorder | Records decisions | |
| Reporter | Reports to class | |
| Reflector | Assesses team performance | |

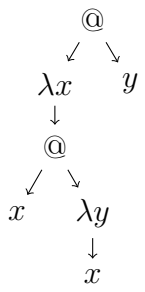
## Reading Syntax and Trees

Consider the following three $\lambda$-calculus expressions and their corresponding trees. Note that the text is identical for all three, but we change some parenthesis.

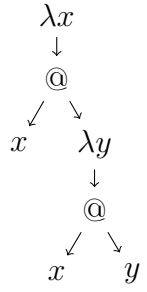
Thing 1 $\lambda x.x(\lambda y.x)y$



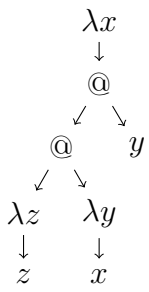

Thing 2 $(\lambda x.x\lambda y.x)y$




Thing 3 $\lambda x.x(\lambda y.xy)$




Thing 4 $\lambda x.(\lambda z.z)(\lambda y.x)y$




**Problem 1)** For each of these trees: how many $\lambda$ abstractions are there? How many application nodes are there?

**Problem 2)** The standard syntax rules for $\lambda$-calculus state that after a $\lambda$, the body of the function goes ``as far as it can.'' How is that reflected in the syntax trees?

**Problem 3)** Thing 1 has an @ with another @ as its left child. What do you think that means? Which of these would you do first during a reduction?

**Problem 4)** The $y$ in Thing 3 has a different status than the $y$ in Thing 1 and Thing 2. How does this different status change the meaning of $y$?

**Problem 5)** Here are three new vocab words: *Reducible Expression* (a.k.a. *redex*), *Weak Head Normal Form* (WHNF), and *Normal Form*. Each one of the Things above has one of these three properties. Speculate about Thing is in which form.

## Do some Reductions!

It is essential that you can do lambda calculus reductions. Have each group member take one of the first four and attempt a reduction. If a varaiable would be $\alpha$-captured, rename the offending $\lambda$. Show your work to each other, and come to a consensus on what the correct answer is.

After that, as a group try the last reduction, but don't spend too much time on it.

1. $(\lambda x.x)\ y$

2. $(\lambda x.x\ z)\ (\lambda y.y)$

3. $(\lambda x.x\ (\lambda x.y))\ (\lambda z.z)$

4. $(\lambda x.(\lambda y.x))\ y\ (\lambda z.z)$

5. $(\lambda x.x\ x)\ (\lambda x.x\ x)$

## Boolean Lambda Terms

In lambda calculus, we can represent a boolean as a function that takes two parameters. A true function returns the first parameter, and a false function returns the second parameter.

6. Write the definitions of $True$ and $False$ as lambda calculus terms.

7. Write the definitions of $and$, $or$, $not$, and $if$.

# Lambda Calculus Activity--- Reflector's Report

| | | |
|---|---|---|
| Manager | Keeps team on track | |
| Recorder | Records decisions | |
| Reporter | Reports to Class | |
| Reflector | Assesses team performance | |

1. What was a strength of your team's performance for this activity?

2. What could you do next time to increase your team's performance?

3. What insights did you have about the activity or your team's interaction today?