

Recitation Notes
Challenges**Challenge 1**

Alonzo Church (1903–1995), inventor of the λ calculus, had an interesting way of writing numbers using only λ expressions. The idea is that each positive integer is represented as the appropriate number of function applications. These are called Church Numerals.

<i>object</i>	<i>Church equivalent</i>
zero	<code>(lambda (f x) x)</code>
one	<code>(lambda (f x) (f x))</code>
two	<code>(lambda (f x) (f (f x)))</code>
inc	<code>(lambda (g f x) (f (g (f x))))</code>

- (A) How might we define `add`, `mult` and `exp` to operate on these unusual numbers?
- (B) How would you define negative numbers? Do your definitions for `add`, `mult` and `exp` still work, or do they need to be modified?

Challenge 2

Given a list l which may or may not contain a circular structure, devise a constant-space algorithm that will always return a value, and in particular will return `true` iff l is circular.

- (A) Your answer may modify l .
- (B) Now, devise a constant-space algorithm that returns the same values, but does not modify l .
- (C) Finally, devise a second constant-space non-mutating algorithm.