SCIENCES
SORBONNE
UNIVERSITÉ

Deactivating ARC in Xcode

Fabrice.Kordon@lip6.fr

LIP6

ARC — Automatic Reference Counting

- Manage memory?
  - Difficult problem
    - Consistency, fragmentation
- Programming Languages and memory management
  - No management
    - Explicit handling of dynamically allocated memory (C)
  - «Hypotheses» to help dealing with fragmentation
    - LISP, Scheme, etc.
  - «Garbage collection»
    - Traversal among the hierarchy of allocated memory (Java)
    - «costly» + non determinism behavior
  - Reference counting
    - To detect absence of owner

ARC — Automatic Reference Counting

Manage memory?

- Difficult problem
  - Consistency, fragmentation

Programming Languages and memory management

- No management
  - Explicit handling
- «Hypotheses» to
  - LISP, Scheme, etc
- «Garbage collection»
  - Traversal among the hierarchy of allocated memory (Java)
  - «costly» + non determinism behavior
- Reference counting
  - To detect absence of owner

Let's see…

… how to deactivate ARC

F. Kordon - Sorbonne Université - CC2018

Deactivate ARC in a project

Deactivate ARC for a given class

F. Kordon - Sorbonne Université - CC2018

As a conclusion…

When ARC is deactivated

- Some instruction can be used + management of dealloc
  - release, retain
  - Very next videos

Of course!

- ARC deactivation available for Objective-C only
- Swift also uses reference counters
  - Behavioral hypotheses to enable explicit management
  - Class notion is well structured

Philosophical question

- What if we mix Swift classes and Objective-C classes without ARC?
  - iOS handles it quite well
  - ARC is deactivated for the Objective-C classes only