Struct VS Class: value vs reference, inheritance, struct auto generate initializers, ARC, attributes in struct is immutable when struct is let while var attributes in class is mutable even the class is declared as let, polymorphism(多态), stack vs heap

viewDidLoad: initializing properties, configuring static content, and setting up views. Called once.
viewWillAppear: tasks that need to be done before every time the view is shown, like refreshing data or UI updates.
viewDidAppear: starting animations or require the view to be fully visible.
viewWillDisappear: stopping tasks, saving progress, or performing cleanup
viewDidDisappear: Final cleanup or stopping services
viewWillLayoutSubviews / viewDidLayoutSubviews: For handling layout adjustments before and after subviews are laid out.

Frame vs. Bound: Defines the view's position and size in its **superview's coordinate system**. VS **own coordinate system**.

DispatchQueue vs. OperationQueue: serial and **concurrent**

Operation objects to encapsulate tasks, enabling dependency management and more granular control over execution order.

Supports concurrent operations with maxConcurrentOperationCount and can cancel or pause operations.

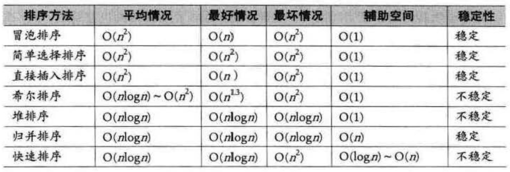
RunLoop.main vs DispatchQueue.main: RunLoop.main for managing timers, event sources, or scheduled tasks. DispatchQueue.main for dispatching work to the main thread, especially UI updates.

Weak: Holds a weak reference that doesn't prevent the object from being deallocated. It's useful to avoid retain cycles, especially in delegate relationships.

Unowned: Similar to weak, but it's used when the reference will never be nil after initialization. If you access an unowned reference after the object is deallocated, it will cause a crash.

@escaping for closures that need to persist beyond the function's scope.

@frozen is an attribute you can apply to enums and structs to indicate that the type's definition is considered "frozen," meaning it will not change in the future



排序方法	平均情况	最好情况	最坏情况	辅助空间	稳定性
冒泡排序	$O(n^2)$	$O(n)$	$O(n^2)$	$O(1)$	稳定
简单选择排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
直接插入排序	$O(n^2)$	$O(n)$	$O(n^2)$	$O(1)$	稳定
希尔排序	$O(n\log n) \sim O(n^2)$	$O(n^{1.3})$	$O(n^2)$	$O(1)$	不稳定
堆排序	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(1)$	不稳定
归并排序	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n)$	稳定
快速排序	$O(n\log n)$	$O(n\log n)$	$O(n^2)$	$O(\log n) \sim O(n)$	不稳定