

每日读书一小时

使用swift实现选择排序算法

```
class SelectionSort {  
  
    func selectionSort(arr: inout [Int]) {  
  
        let length = arr.count // 数组长度  
  
        for i in 0 ..< length - 1 { // 遍历数组  
            var min = i // 假定数组最小元素下标  
            for j in i + 1 ..< length { // 寻找未排序数组中最小元素的下标  
                if arr[min] > arr[j] { // 如果当前最小元素 大于 未排序数组中的最小元素  
                    min = j // 交换最小元素下标  
                }  
            }  
            arr.swapAt(min, i) // 将最小元素放到已排序数组末尾  
        }  
    }  
}
```

文字说明

选择排序 实现思路是 首先从数组中找到最小的元素放到排序起始位置，
然后从剩余元素中继续寻找最小元素并放到已排序数组的末尾，直到排序完成
如果某个元素处于正确的位置上，则不会被移动

通过用swift语言实现选择排序复习了swift的 for loop 一开始犯的错是使用了...进行数组循环导致数组越界 ... 是闭区间因此包含第一个元素和最后一个元素。

同时学到了一个新的API collection的swapAt方法 函数签名 `func swapAt(_ i: Int, _ j: Int)`

作用是

```
/// Exchanges the values at the specified indices of the collection.
///
/// Both parameters must be valid indices of the collection that are not
/// equal to `endIndex`. Calling `swapAt(_:_:)` with the same index as both
/// `i` and `j` has no effect.
///
/// - Parameters:
/// - i: The index of the first value to swap.
/// - j: The index of the second value to swap.
```

翻译成中文就是 交换集合中两个元素的位置。