```
    subscript(i: Int) -> ItemType {get}
}

struct QueueType<T>: ExtraInformation {
    var items = [T]()
    mutating func insert(item: T) {
        items.append(item)
    }
    mutating func delete() {
        items.remove(at: 0)
    }

    //comformance to the ExtraInformation Protocol
    typealias ItemType = T
    var count: Int {
        return items.count
    }
    subscript(i: Int) -> T {
        return items[i]
    }
}

var queueOfData = QueueType<Int>()
queueOfData.insert(item: 100)
queueOfData.insert(item: 200)
queueOfData.insert(item: 300)
queueOfData.insert(item: 400)
queueOfData.insert(item: 500)
print("陣列中有\(queueOfData.count)個元素")
for i in queueOfData.items {
    print("\(i) ", terminator: "")
}
print("\n")

queueOfData.delete()
print("刪除100後，陣列中有\(queueOfData.count)個元素")
for i in queueOfData.items {
    print("\(i) ", terminator: "")
}
```



```
print("\n")

queueOfData.insert(item: 600)
print("加入600後，陣列中有\(queueOfData.count)個元素")
for i in queueOfData.items {
    print("\(i) ", terminator: "")
}
print("\n")

print("queueOfData[2] = \(queueOfData[2])")
print("")
```

輸出結果如同上述的整數佇列。

您也可以依樣畫葫蘆，以字串或浮點數佇列加以驗證之，這就當做自我練習題。

自我練習題

1. 以泛型的方式實作堆疊的加入與刪除。

2. 以下程式皆有些許的 bugs，請你加以 debug，順便測驗你對本章了解的程度。

(a)

```
struct QueueString {
    var items = [String]()
    mutating func insert(item: String) {
        items.append(item)
    }
    mutating func delete() {
        items.remove(0)
    }
}

var queueOfString = QueueOfString()
queueOfString.insert(Guava)
queueOfString.insert(Apple)
queueOfString.insert(Orange)
queueOfString.insert(Kiwi)
queueOfString.insert(Mango)
print("Queue has following elements: ")
for i in queueOfString.items {
    print("\(i) ")
}
print("")

queueOfString.delete()
println("Queue has following elements: ")
for i in queueOfString.items {
    print("\(i) ", terminator: "")
}
print("")

queueOfString.delete()
queueOfString.delete()
print("Queue has following elements: ")
```



```
for i in queueOfString.items {
    print("\(i) ", terminator: "")
}
print("")

queueOfString.insert(Pearl)
println("Queue has following elements: ")
for i in queueOfString.items {
    print("\(i) ", terminator())
}
print("")
```

(b)

```
func searchData<T>(array: [T], valueToSearch : T) -> Int? {
    for (index, value) in enumerate(array) {
        if value == valueToSearch {
            return index
        }
    }
    return nil
}

let arrayofStrings = ["Apple", "Guava", "Banana", "Kiwi", "Orange"]
let found = searchData(arrayofStrings, "Kiwi")
print("The index of Kiwi is \(found)")
let found2 = searchData(arrayofStrings, "Pineapple")
print("The index of Pineapple is \(found2)")

let arrayOfInt = [11, 22, 33, 44, 55]
let found3 = searchData(arrayOfInt, 55)
print("\nThe index of 55 is \(found3)")
let found4 = searchData(arrayOfInt, 66)
print("The index of 66 is \(found4)")

let arrayOfDouble = [11.1, 22.2, 33.3, 44.4, 55.5]
let found5 = searchData(arrayOfDouble, 22.2)
print("\nThe index of 22.2 is \(found5)")
let found6 = searchData(arrayOfDouble, 66.6)
print("The index of 66.6 is \(found6)")
```

(c)

```
//generic bubble sorting
func bubbleSort<T>(arr: [T]) {
    var flag: Bool
    var i: Int, j: Int

    for i=0; i<arr.count-1; i++ {
        flag = false
        for j=0; j<arr.count-i-1; j++ {
            if arr[j] > arr[j+1] {
                flag = true
                let temp = arr[j]
                arr[j] = arr[j+1]
                arr[j+1] = temp
            }
        }
        if flag == false {
            break
        }
    }
}

var arrOfString = ["Mary", "Peter", "Amy", "Jennifer", "Nancy", "Bright"]
print("\nBefore sorted: ")
for i in arrOfString {
    print("\(i) ", terminator: "")
}

bubbleSort(arrOfString)

println("\nAfter sorted: ")
for j in arrOfString {
    print("\(j) ", terminator: "")
}

var arrOfInt = [12, 9, 8, 35, 2, 10, 17, 9]
print ("\n\nBefore sorted: ")
for i in arrOfInt {
    print("\(i) ", terminator: "")
}

bubbleSort(arrOfInt)

print("\nAfter sorted: ")
```



```
for j in arrOfInt {
    print("\(j) ", terminator: "")
}

var arrOfDouble = [1.2, 2.9, 1.8, 3.5, 2.1, 1.1, 0.17, 0.9]
print("\n\nBefore sorted: ")
for i in arrOfDouble {
    print("\(i) ", terminator: "")
}

bubbleSort(arrOfDouble)

println("\nAfter sorted: ")
for j in arrOfDouble {
    print("\(j) ", terminator: "")
}
print("")
```

3. 請修改本章的最後一個範例，將字串或浮點數加入於佇列，並加以驗證其結果。