

# STL

## STL (standard template library) 標準函式庫

### SLT基本概念

#### 大致上分為三種

- 容器(container)

容器是用來管理某一類對象的集合。C++ 提供了各種不同類型的容器，比如 deque、list、vector、map 等

- 演算法(algorithm)

演算法作用於容器。它們提供了執行各種操作的方式，包括對容器內容執行初始化、排序、搜索和轉換等操作。

- 迭代器(iterator)

迭代器用於遍歷對象集合的元素。這些集合可能是容器，也可能是容器的子集

### SLT標頭檔

- |               |             |                |              |           |
|---------------|-------------|----------------|--------------|-----------|
| • <algorithm> | • <deque>   | • <functional> | • <iterator> |           |
| • <list>      | • <map>     | • <memory>     | • <numeric>  |           |
| • <stack>     | • <utility> | • <vector>     | • <set>      | • <queue> |

### 重要特點

- 不用額外安裝
- 資料結構與演算法分離
- 具高功能、高移植性、跨平台

## 容器

- 循序式容器

容器元素的位置是由進入容器的時機和地點來決定

循序式容器是一種線性的資料儲存容器 分成vector、deque與list等三種

- 關聯式容器

容器內已有規則 進入容器的元素位置不是由進入時間和地點決定

關聯式容器為非線性儲存容器 可以快速搜尋資料 可以用於儲存資料的集合 與鍵值與儲存值(key與value)配對的資料結構

分成set、multiset、map與multimap等四種

## 迭代器

迭代器可視為一個指標 對指標的操作基本上都可以對迭代器操作

事實上迭代器是一個類別 這個類別封裝一個指標

## vector容器

### 特性

- 動態陣列:

vector以陣列為儲存資料的容器，系統會動態增加或減少陣列的大小，不須事先宣告vector的大小，可以利用索引值直接存取指定的資料

- 優點
  - 宣告時不用事先確定大小
  - 支援random access
- 缺點
  - 內部刪除效率差

函式分類	函式	功能
迭代器函式	begin()	回傳vector的第一個元素
	end()	回傳vector的最後一個元素的下一個，這個元素不存在，通常用於表示資料已經到達結尾
新增或刪除元素函式	push_back(x)	將x加到vector最後一個元素。
	pop_back()	刪除最後一個元素。
	insert(pos,x)	在vector的pos位置插入元素x。
	erase(pos)	刪除vector的pos位置的元素。
	clear()	刪除vector中所有元素
讀取元素函式	at(i)或[i]	讀取vector第i個元素，at(i)在i值超出邊界時會發出錯誤訊息，而[i]在i值超出邊界時不會發出錯誤訊息。
	front()	讀取vector第一個元素
	back()	讀取vector最後一個元素
儲存空間函式	empty()	檢查vector是否是空的，若是空的，回傳true，否則回傳false。
	size()	回傳vector目前儲存元素的個數