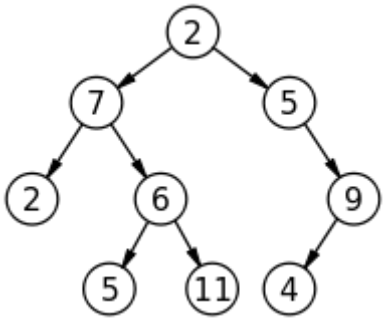
WIKIPEDIA

数据结构

维基百科，自由的百科全书

在计算机科学中，**数据结构**（英語：data structure）是计算机中存储、组织数据的方式。



二叉树是数据结构的一种类型

数据结构意味着介面或封装：一个数据结构可被视为两个函数之间的介面，或者是由数据类型联合组成的存储内容的访问方法封装。

大多数数据结构都由数列、记录、可辨识联合、引用等基本类型构成。举例而言，可為空的引用（nullable reference）是引用与可辨识联合的结合体，而最简单的链式结构链表则是由记录与可空引用构成。

数据结构可透过程式语言所提供的数据类型、引用及其他操作加以实现。一个设计良好的数据结构，应该在尽可能使用较少的时间与空间资源的前提下，支援各種程式執行。

不同种类的数据结构适合不同种类的应用，部分資料結構甚至是為了解決特定問題而設計出來的。例如B树即為加快樹狀結構存取速度而設計的資料結構，常被應用在資料庫和檔案系統上。

正確的数据结构選擇可以提高演算法的效率（請參考演算法效率）。在電腦程式设计的過程中，选择适当的数据结构是一項重要工作。许多大型系统的編寫经验顯示，程式設計的困难程度与最终成果的质量与表现，取决于是否选择了最適合的数据结构。

系统架構的关键因素是数据结构而非算法的見解，导致了多种形式化的设计方法与编程语言的出现。绝大多数的语言都带有某种程度上的模块化思想，透过将数据结构的具体实现封装隐藏于使用者介面之后的方法，来让不同的应用程序能够安全地重用这些数据结构。C++、Java、Python等面向对象的编程语言可使用类 (计算机科学)来達到這個目的。

因为数据结构概念的普及，现代编程语言及其API中都包含了多种預設的数据结构，例如 C++ 标准模板库中的容器、Java集合框架以及微软的.NET Framework。

常见的数据结构

- 堆疊（Stack）
- 佇列（Queue）
- 陣列（Array）
- 链表（Linked List）
- 樹（Tree）
- 圖（Graph）
- 堆積（Heap）
- 雜湊表（Hash table）

参考文献

外部链接

- 《算法与数据结构词典》中的描述 (http://nist.gov/dads/)
- http://www.cse.unr.edu/~bebis/CS308/
- Bruno R. Preiss，面向对象程序设计的数据类型与算法模型：C++ (http://www.brpreiss.com/books/opus4)、Java (http://www.brpreiss.com/books/opus5)、C# (http://www.brpreiss.com/books/opus6)、Python (http://www.brpreiss.com/books/opus7)、Ruby (http://www.brpreiss.com/books/opus8)
