

离线算法简介

本章将介绍介绍离线算法（Offline Algorithm）的思想、常见算法及优化。



离线算法是基于「**求解前已知所有数据**」这一假设来设计的，适用于有多组询问的题目。相对的还有 [在线算法](#)（Online Algorithm）。

例如 [选择排序](#) 必须知道数组的全局最小元素才能执行，所以是离线算法，而 [插入排序](#) 可以动态接收数据进行排序，不强制要求执行前已知全部数据，所以是在线算法。

对于相同的问题，在设计难度等方面，离线算法往往优于在线算法。为了阻止选手使用离线算法，有时题目会使用「强制在线」的方式，常见的有需要前一个询问的答案才能得到下一个询问的参数（[交互题](#) 与 [通信题](#) 也属于此类）。

离线算法的常见思路包括将询问统一求解（如 [CDQ 分治](#)）、通过一个询问的答案求出另外相似询问的答案（如 [整体二分](#) 和 [莫队算法](#)）等。

由于离线算法是一种思想而并不是某种具体的算法，因此它会搭配各种各样的数据结构或算法一起使用，与之相关的题目种类也更为繁杂。

本页面最近更新：2025/2/8 15:15:39，[更新历史](#)

发现错误？想一起完善？ [在 GitHub 上编辑此页！](#)

本页面贡献者：[lr1d](#), [Oracynx](#), [Tiphereth-A](#)

© 本页面的全部内容 [在 CC BY-SA 4.0 和 SATA 协议之条款下](#) 提供，附加条款亦可能应用

