



1 初识算法

1.1 算法无处不在

当我们听到“算法”这个词时，很自然地会想到数学。然而实际上，许多算法并不涉及复杂数学，而是更多地依赖于基本逻辑，这些逻辑在我们的日常生活中处处可见。

在正式探讨算法之前，有一个有趣的事实值得分享：你已经在不知不觉中学会了许多算法，并习惯将它们应用到日常生活中了。下面，我将举几个具体例子来证实这一点。

C++ 中的仿函数（Functor）是一种特殊的对象，它可以像函数一样被调用，但实际上是一个类的实例。仿函数可以通过重载函数调用运算符 `operator()` 来实现，从而使得该对象可以像函数一样被调用。仿函数主要用于在算法中作为操作符使用，例如在 STL（标准模板库）中的各种算法中。通过使用仿函数，可以将操作抽象出来，使得算法更加灵活，可以适用于不同的数据类型或者自定义的操作。

```
1  #include <iostream>
2
3  // 定义一个加法仿函数
4  class AddFunctor {
5  public:
6      int operator()(int a, int b) {
7          return a + b;
8      }
9  };
10
11 int main() {
12     AddFunctor adder; // 创建仿函数对象
13
14     int result = adder(3, 5); // 使用仿函数调用
15     std::cout << "Result:" << result << std::endl;
16
17     return 0;
18 }
```