前言：

学习编程语言的数据结构是枯燥的，但是数据结构是算法的基础。程序的本质，就是帮助人完成一下复杂且需要大量重复的事；这也是计算机擅长领域。在现实生活中来说，大多数互联网应用，无非就是把原先一些非互联网的业务，挪到互联网上运作，因为互联网有庞大的用户群。如电商、打车、外卖、等都是。还有一部分软件，是将一些传统的，需要手动完成的工作挪到计算机上完成，以提升工作效率，如绘图软件等。作为程序员，大多数时候，我们就是在与存储在计算机上的数据打交道，因为程序员的工作无非就是，把处理好的数据，通过各种方式展现给用户。

为什么要学习数据结构？

在正式讨论这个问题之前，我想先抛出一个程序相关的常见问题：  
“假设你需要检索到一个约有20000页内容的PDF文件的第15000页中的内容，你需要怎么完成？”

注意：在计算机领域，不会像现实世界那么直观，计算机需要找到你需要的那部分数据，然后将那部分数据扫描到内存中，然后供程序使用。

如果你尝试随机的抽取，或者顺序查看的方式，这将会耗费大量时间。

但如果你尝试另外一种方式（这其实跟我们现实生活中的思考方式查不多）：  
1.先去到第1000页数据存放的位置（注意，在计算机中硬盘或内存中的数据可能不是连续存放的，所以这个位置可能是你通过程序随机控制的，假设我们走到了第10000页内容在计算机硬盘中存放的位置，也许不是第1000页，也许你走到了9000页，也行，不要那么死板昂。）

1. 如果你需要的数据不是这一部分，但第10000页中的数据的roll number（这个是分配到这部分数据的一个用于标注这一部分数据类似序号的东西，并不能与现实世界中真实的页码匹配）都小于你要的那部分数据的roll number。
2. 我们已经知道，我们需要的数据在10000页数据的roll number的后面，所以现在我们只需要从后面这部分数据下手就可以了。
3. 我们进一步缩小范围，假设现在我们一下走到了16000页内容存放的位置，此时我们知道roll number又大于了我们需要内容的roll number。
4. 现在我们知道我们需要搜索的内容在16000页内容roll number的前面。现在我们就需要扫描前面一部分数据，然后找出第15000页内容的数据即可。

也许你会很奇怪，既然知道roll number那为什么不直接定位到这一部分内容的内存，直接读取呢？原因是这样的：

1. 这个roll number并不代表真实的存储地址。
2. 先前我们提到过，如果内存地址是连续的还好，你可以像读取数组中的数据那样，直接通过内存地址来读取数据，如果数据不是连续的，我们必须要通过某一内存地址指向的数据中记录的：上一部分内容记录在哪个内存地址当中，下一部分内容记录在那个内存地址中。

但不管怎么说：你已经在用二叉搜索树这种数据结构的知识来解决问题了。

所以学习数据结构的好处就在于，你可以用数据结构的知识，有效的组织和存储你的数据，这样你才可使用计算机高效的解决一些复杂且需要大量重复的问题。