02.1 - Searching in Logarithmic Time

线性搜索（Linear Search）与二分搜索（Binary Search）的比较

二分搜索：

- 时间 $O(\log n)$
- 空间 $O(\log n)$

线性搜索：

- 时间 $O(n)$
- 空间 $O(1)$

二分搜索（Binary Search）

假设我们寻找 x，二分比较中间的元素与 x 的大小；直至子数组的长度为 1，此时检查元素是否为 x

```
Procedure BinarySearch(x, i, j):
  If i = j then
    If x = A[i] return yes
    If x ≠ A[i] return no
  Else
    If x = A[(i + j) / 2] return yes
    If x < A[(i + j) / 2] return BinarySearch(x, i, (i + j) / 2 - 1)
    If x > A[(i + j) / 2] return BinarySearch(x, (i + j) / 2 + 1, j)
```

二分搜索的运行时间

- 所有操作都在常数时间内完成，并且只有一个常数数量的非比较操作
- 运行时间主要通过比较操作的次数来衡量，每次调用二分搜索过程最多进行4次比较（伪代码中 2、6、7、8 行）

所以，二分搜索的比较次数是

$$
\begin{aligned}
T(n) &\leq T(\frac{n}{2}) + 4 \\
&\leq T(\frac{n}{4}) + 4 \\
&\leq T(\frac{n}{8}) + 12 \\
&\dots \\
&\leq T\left(\frac{n}{2^{\log_2 n - 1}}\right) + 4(\log_2 n - 1) \\
&= T\left(\frac{2}{\frac{n}{2}}\right) + 4(\log_2 n - 1) \\
&= T(2) + 4(\log_2 n - 1) \\
&\leq 4 + 4(\log_2 n - 1) \\
&= 4\log_2 n
\end{aligned}
$$

其中，$\log_2 n - 1$ 是递归的次数：对于一个拥有 n 个元素的数列，我们不断进行二分，需要 $\log_2 n$ 次；而最后一次我们直接进行比较，不再进行二分，因此减少一次

使用数学归纳法（Mathematical Induction）来证明二分搜索（Binary Search）的运行时间为对数时间

对于基本情况 $n=2$，显然 $T(2) \leq 4\log_2 2$
假设对于所有小于 n 的输入规模都成立：$T(\frac{n}{2}) \leq 4\log_2 \frac{n}{2}$
我们需要证明对于 n 也成立：$T(n) \leq 4\log_2 n$

根据递推关系

$$T(n) \leq T\left(\frac{n}{2}\right) + 4$$

代入归纳假设

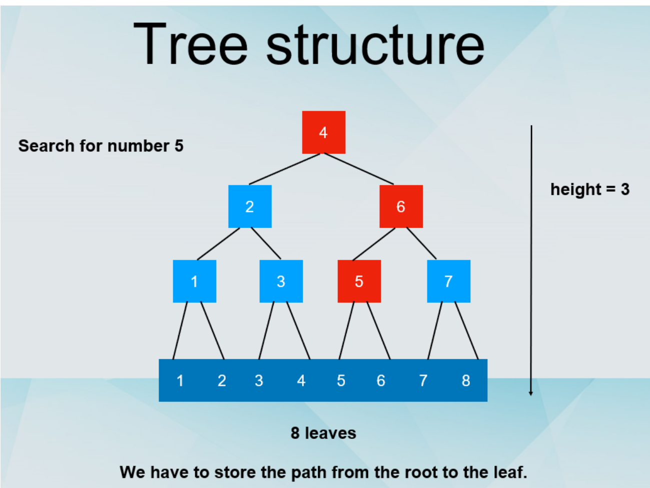
$$T(n) \leq 4\log_2 \frac{n}{2} + 4$$
$$4\log_2\left(\frac{n}{2}\right) = 4\left(\log_2 n - \log_2 2\right) = 4\log_2 n - 4$$

因此

$$T(n) \leq 4\log_2 n - 4 + 4 = 4\log_2 n$$

二分搜索的内存需求（Memory requirements of Binary Search）

1. 输入部分使用的内存：
    - 存储数组需要 n 个单位的内存（假设数组长度为 n）
    - 存储目标数 x 需要 1 个单位的内存
    - 因此，输入部分的总内存需求为：$n+1$
2. 辅助内存（Auxiliary memory）：
    - 辅助内存是算法在执行过程中额外使用的内存
    - 由于二分搜索是递归算法，它在执行过程中会多次调用自身
    - 这些递归调用需要保持在内存中，直到所有递归调用完成
3. 递归调用的数量：
    - 二分搜索每次将问题规模减半，因此递归调用的深度为对数级别
    - 具体来说，递归调用的总次数为 $O(\log n)$



因此，我们需要 $O(\log n)$ 的空间