

Homework 3

August 27, 2026

1 Asymptotic Notation

1. Prove that $f(n) = 5n^2 + 3n$ is $O(n^2)$.

Solution. Take $c = 8$ and $n_0 = 1$. For $n \geq 1$, $3n \leq 3n^2$, so $5n^2 + 3n \leq 8n^2$, which satisfies the definition of $O(n^2)$.

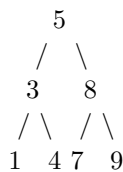
2. Rank the following functions by growth rate, slowest to fastest: $n \log n$, 2^n , $n!$, $\sqrt{n}$.

Solution. $\sqrt{n} < n \log n < 2^n < n!$.

2 Binary Search Trees

1. Insert the keys 5, 3, 8, 1, 4, 7, 9 in that order into an initially empty binary search tree. Draw the resulting tree.

Solution. Each key after the first is placed by comparing against the root and recursing left (smaller) or right (larger):



2. What is the time complexity of search, insert, and delete on a balanced BST with n nodes? What about an unbalanced one in the worst case?

Solution. Balanced: $O(\log n)$ for all three operations. In the worst case (a degenerate, list-shaped tree) all three degrade to $O(n)$.

3 Code Listing

Listing 1: Iterative binary search

```
1 def binary_search(a, target):
2     lo, hi = 0, len(a) - 1
3     while lo <= hi:
4         mid = (lo + hi) // 2
5         if a[mid] == target:
6             return mid
7         elif a[mid] < target:
8             lo = mid + 1
9         else:
10            hi = mid - 1
11    return -1
```