Notes for ECE 36800 - Data Structures

Zeke Ulrich

January 16, 2025

Contents

Course Description

Provides insight into the use of data structures. Topics include stacks, queues and lists, trees, graphs, sorting, searching, and hashing.

Introduction

An algorithm is simply a method to solve a class of problems. There are algorithms to make toast, to solve a Rubik's cube, and to plot the optimal path for a mailmain. The two most important qualities of any algorithm are effectiveness and secondly efficiency. Our solution must work, preferably in as little time as possible. A working algorithm is no good if the heat death of the universe occurs before it finds a solution. We describe the cost of an algorithm using Big O notation. As an example, consider listing 1 If each operation has cost C_i, then

```
int total = 0; // C_1
for (int i = 0; i++; i <= n){ // C_2
    total = total + i; // C_3
}
return total; // C_4
```

Figure 1: Sum of first n numbers

the total cost of the program is

$$C_1 + C_2(n+1) + C_3 n + C_4 = n(C_2 + C_3) + (C_1 + C_4). \qquad (1)$$

Big O notation considers only the largest power of n, so the Big O complexity for this algorithm would be $O(n)$.

A data structure is an organization of information for ease of manipulation. For example, a dictionary, a checkout line, and an org chart.