DS.I.1

CSE 373: Data Structures and Algorithms

Autumn Quarter 2000

Instructor: Prof. Linda Shapiro
Email: shapiro@cs.washington.edu

Tas: Valentin Razmov
Ann Li
Karen Liu

Purpose: to study the fundamental data structures and algorithms used in computer science.

- **Data Structures**
- **Algorithm Analysis**
- **Applications**

DS.I.2

Text: Data Structures and Algorithm Analysis in C

Topics to be Covered

• Introduction	(Ch 1)
• Algorithm Analysis	(Ch 2)
• Review of Lists, Stacks, Queues	(Ch 3)
• Trees and Search Trees	(Ch 4)
• Hashing	(Ch 5)
• Priority Queues: Heaps	(Ch 6)
• Disjoint Sets: Union-Find	(Ch 8)
• Graph Algorithms	(Ch 9)

Applications

Computer Graphics
Computer Vision
Artificial Intelligence
Databases

DS.I.3

INTRODUCTION

Chapter 1 Overview

- **Motivation for Good Algorithm Design**
- **Math Review (for use in reading proofs)**
- **Proofs by Induction, Counterexample, Contradiction**
- **Recursion**

What is an algorithm ?

What would be a good algorithm for achieving tulips in my garden next spring?

DS.I.4

An algorithm to achieve X is a procedure that:

1. Halts
2. Correctly achieves X

Procedure for computing N / 5 for integer N

```
count = 0;
do {
    count = count + 1;
    N = N - 5;
    }
until (N <= 0)
```

Does it always halt?

Does it correctly compute N / 5?

DS.I.5

What is a data structure ?

Informal Definition: a method of organizing data

Examples?

Formal Definition: an abstract data type (ADT)

In C:

- structs
- functions that operate on them

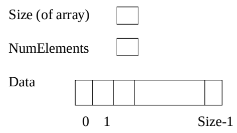
In C++:

- classes
- methods

DS.I.6

Example: VectorArray

Conceptual:



What abstract operations are needed?

DS.I.7

Iteration, Recursion, Induction

1. Write an iterative function to find the sum of the first num elements of a VectorArray stored in array v.

```
int sumit ( int v[ ], int num)
  {
  int sum;
  sum = 0;
  for (                                    )
       sum =                               ;
  return sum;
  }
```

DS.I.8

2. Write a **recursive** function to find the sum of the first num elements of a VectorArray stored in array v.

```
int sumit ( int v[ ], int num)
  {



  }
```

Strengths of recursive approach:
- simplifies code
- can be proven correct

Weaknesses:
- slower than iteration
- uses more memory for the stack

DS.I.9

Principle of Mathematical Induction

Let P(c) be true for small constant c >= 0.

Suppose whenever P(k) is true, so is P(k+1).

Then P(n) is true for all n >= 0.

Ex. 1.10a Prove by induction that

$$\sum_{i=1}^{N} (2i - 1) = N^2.$$

Basis: N=1

Inductive Hypothesis:

Induction Step:

DS.I.10

```
int sumit (int v[ ], int num)
  {
  if (num == 0) return 0;
  else return v[num-1] + sumit(v,num-1);
  }
```

Prove by induction that sumit(v,n) correctly returns the sum of the first n elements of array v, n≥0.

Basis: If n=0,

Inductive Hypothesis: Assume sumit(v,k) ...

Inductive Step: sumit(v,k+1)