

CSCI 333 Data Structures

Introduction
14 August 2002

On-Line Information

- <http://www.cs.unca.edu/csci/333>
- [Course syllabus](#)
- [Course schedule \(evolving\)](#)
- [Last year's CSCI 333](#)

Local Resources

- Linux C++ (G++)
- PC Software
 - MS Visual Studio 6.0
 - [MS Visual Studio .NET](#)
 - [MS Windows XP Pro](#)
 - And other operating systems

Notes with the dark blue background
were prepared by the textbook author

Clifford A. Shaffer
Department of Computer Science
Virginia Tech
Copyright © 2000, 2001

The Need for Data Structures

Data structures organize data
⇒ more efficient programs.
More powerful computers ⇒ more complex
applications.
More complex applications demand more
calculations.
Complex computing tasks are unlike our
everyday experience.

Organizing Data

Any organization for a collection of records
can be searched, processed in any order,
or modified.
The choice of data structure and algorithm
can make the difference between a
program running in a few seconds or many
days.

Efficiency

A solution is said to be efficient if it solves the problem within its resource constraints.

- Space
- Time
- The cost of a solution is the amount of resources that the solution consumes.

Selecting a Data Structure

Select a data structure as follows:

1. Analyze the problem to determine the resource constraints a solution must meet.
2. Determine the basic operations that must be supported. Quantify the resource constraints for each operation.
3. Select the data structure that best meets these requirements.

Data Structure Philosophy

Each data structure has costs and benefits.

Rarely is one data structure better than another in all situations.

A data structure requires:

- space for each data item it stores,
- time to perform each basic operation,
- programming effort.

Abstract Data Types

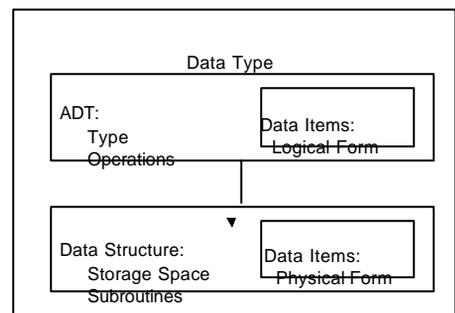
Abstract Data Type (ADT): a definition for a data type solely in terms of a set of values and a set of operations on that data type.

Each ADT operation is defined by its inputs and outputs.

Encapsulation: Hide implementation details.

Data Structure

- A data structure is the physical implementation of an ADT.
 - Each operation associated with the ADT is implemented by one or more subroutines in the implementation.
- Data structure usually refers to an organization for data in main memory.
- File structure is an organization for data on peripheral storage, such as a disk drive.



Problems

- Problem: a task to be performed.
 - Best thought of as inputs and matching outputs.
 - Problem definition should include constraints on the resources that may be consumed by any acceptable solution.

Algorithms and Programs

Algorithm: a method or a process followed to solve a problem.

- A recipe.

An algorithm takes the input to a problem (function) and transforms it to the output.

- A mapping of input to output.

A problem can have many algorithms.

Algorithm Properties

An algorithm possesses the following properties:

- It must be correct.
- It must be composed of a series of concrete steps.
- There can be no ambiguity as to which step will be performed next.
- It must be composed of a finite number of steps.
- It must terminate.

A computer program is an instance, or concrete representation, for an algorithm in some programming language.