

Data Structures and Algorithms

Practical Illustration

- A simple Paint file
- Working on a spreadsheet file
-

Data Structures

- Data structure is the structural representation of logical relationships between elements of data.
- Thus data structure is a way of organizing data items by considering its relationship to each other.
- Constitute building blocks of programs

- Simplistic Denotation:
- **Algorithm + Data Structure = Program**

- Considering accessing methods with correct degree of associativity
- Intentional and structured form of “divide and conquer” approach
- Organized Data + operations = Data Structure

- Arrangement of a collection of data items
- Operations on arranged data
- Efficient utilization

Key components

- CPU
- Main memory
- Secondary memory
- What are their roles?

- Illustration. Coordination
- Data files mainly stored in secondary memory
- File or application retrieved from storage to main memory
- Also organized in main memory for efficient utilization
- Data structures/arrangement in main memory

Related/confusing terminologies

- Database
- Relational database
- Operational vs legacy/historical data
- Data warehouse
- Data mining (algorithms)
- Big data

Data structure types

- Physical. Defines organization of memory
 - Array (Heaps and stacks)
 - Linked List
 - Dynamic.
 - Collection of nodes, containing data linked to next node.
 - Variable length
 - (Created in heaps)

- Logical
 - Stacks
 - Queues
 - Trees
 - Graph
 - Hash Table

- Physical aspect of the data structure involves data storage in memory
- Logical: used in application/algorithms
 - Could be linear, non-linear or tabular with some uniqueness

PROGRESSIVE THREE-STEP-WISE REFINEMENT METHODS

1. Mathematical Model (Informal Algorithm)
2. Formal Language (Pseudo-language Program)
3. Data Structures (By a programming language)

E.g. C/C++/...

ALGORITHM

- The step-by-step, finite sequence of instruction to solve a well-defined computational program
- Relationship between algorithm and optimization (more on this later)

ALGORITHM DESIGN

- STEP-WISE REFINEMENT METHODS
 - Top-down
 - Bottom-UP

ANALYSIS OF ALGORITHM

- Predictability and correctness
- Tracing steps involved
 - Reading , checking logic correctness,
- Testing with some data
- Simplest not always the best.

- Settling on a particular algorithm depends on following performance analysis and measurements:
 - Space complexity
 - Time Complexity

Space complexity

- The amount of memory a program needs to run to completion.
- Rationale for studying/measuring:
 - Specification and allocation of memory to a program in a multi user system
 - Advanced knowledge of sufficiency of memory
 - Several possible solutions with different space requirements
 - Can estimate size of largest problem a program can solve

Space needed by an algorithm

- Instruction space: Space needed to store the executable version of the program. It is fixed.
- Data space : Space needed to store all constants, variable values. It has two sub components:

Questions