Programming and Data Structures

CS2700 and CS2710

Rupesh Nasre.
rupesh@iitm.ac.in

TAs: Maneesh, Sidharth, Deepak, Deepak,
Ananya, Brintha, Niket, Omji, Pranjal.

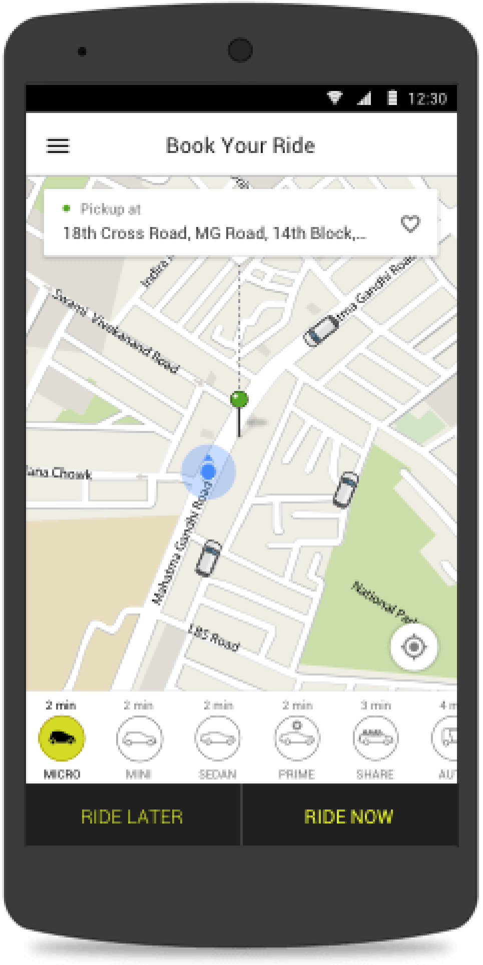
Course webpage: ~rupesh/teaching/pds/jul19
Moodle: https://courses.iitm.ac.in/course/view.php?id=5107

July 2019

Placement in Computer Science

- CS1100: Coding
- CS1200: Proofs, Counting
- CS2200: Computation Theory
- CS2300: Overview of Digital World
- CS2600: Hardware
- **CS2700: Efficient Implementation**
- CS2800: Algorithms
- CS3100: Ways of Programming
- CS3300: Translation (Programmer and Machine)
- CS3500: Resource Management (User and Machine)

Relevance, with an example

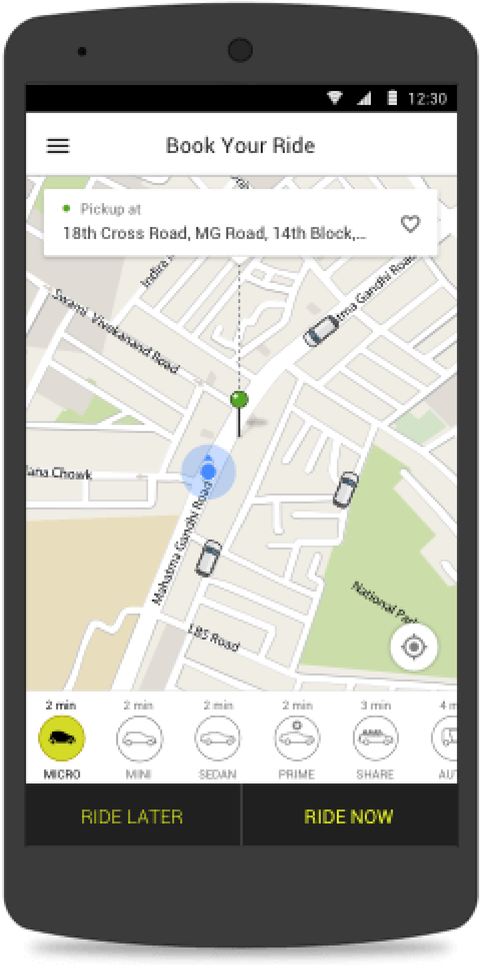


- DM: How many ways can you go from IIT to Central?
- LMC: Can you compute *blah* using a smartphone?
- CO/CA: How to build a smartphone hardware?
- **PDS: How to keep track of cabs such that a passenger can query the nearest cabs efficiently?**
- Algo: How to compute the shortest path from IITM to Central?
- OS: How to give a higher priority to an incoming call?
- Compilers: Translating an app or OS to machine code
- Networks: How does a phone call work?
- DBMS: How to store and retrieve world-map data relevant to the user?

Do not decouple these subjects (especially Algorithms and Data Structures).
They go hand-in-hand, but we emphasize on one hand at a time.

Image source: Google images

More Examples



Pickup at
18th Cross Road, MG Road, 14th Block,...

- How to keep track of cabs such that a passenger can query the nearest cabs efficiently?
- How should I store addresses such that I can show suggestions as user types a destination address?
- How should I categorize cabs such that the passenger is able to view Micro, Mini, Prime, Sharing options quickly?
- I should be able to identify quickly if there are multiple sharers nearby.
- When a cab moves, how should I store the data such that the graphics rendering uses only the diff rather than displaying the complete screen again?
- How should I store previous rides such that I am able to find an approximate cost for this journey prior to booking?

Data structures get more important whenever there is more data, … and more types of data.

Image source: Google images

Misconceptions

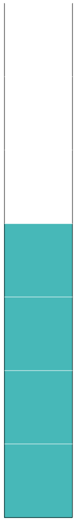
- Data structures get created when we use `struct`.
- Data structures need pointers.
- An algorithm must use a specific data structure.
- C++ has more data structures than C.
- There are in total seven data structures.
- Union-Find is the best data structure.

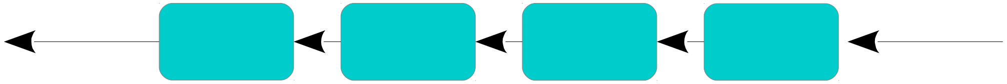
What is it?

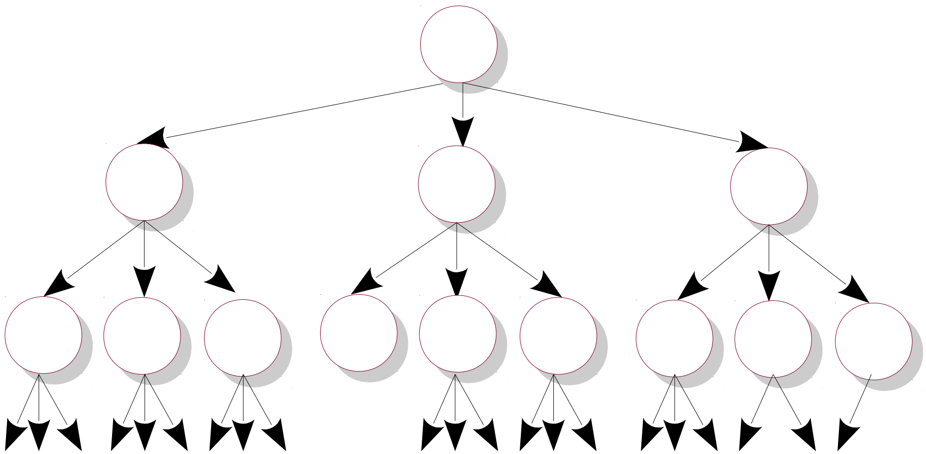
- **Data Structures** is about organizing data such that its storage and retrieval improve the efficiency of the algorithms using it.
- A data structure may be used by multiple algorithms.
- An algorithm may use multiple data structures simultaneously.
- An algorithm may use different data structures and achieve the same computation.

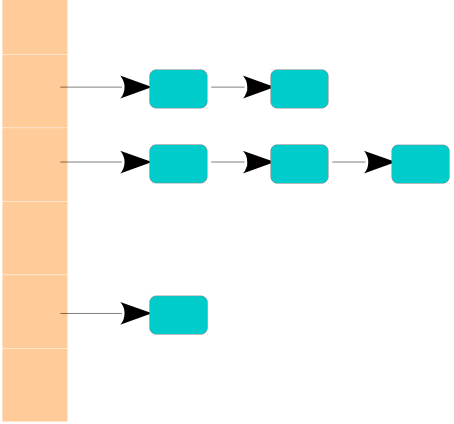
Core and Standard

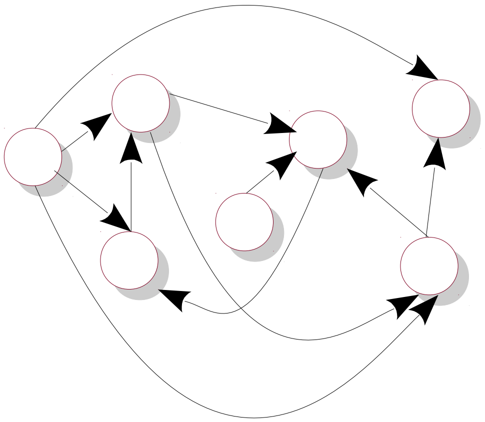
- Array
- Linked List
  - Stack
  - Queue
- Tree
  - Binary Tree
  - Binary Search Tree
  - Heap
  - …
- Hash Table
- Graph











Learning Outcomes

- **Choose efficient data structures and apply them to solve problems.**
- Design correct programs to solve problems.
- Analyze the efficiency of programs based on time complexity.
- Prove the correctness of a program using loop invariants, pre-conditions and post-conditions in programs.

Logistics

- Theory: CS26
  - C slot (Mon 10, Tue 9, Wed 8, Fri 13)
- Lab: DCF
  - R slot (Wed 14)
  - Hackerrank submissions
- Prerequisite: CS1200 Discrete Mathematics
- It is your responsibility to get subscribed to moodle (id=5107 and 4606)
  - Self-enrolment is enabled.
- Attendance: Standard institute rules apply.
- Evaluation

Lab

- Assignments
  - Problems on Hackerrank
  - Mostly C (a little bit of C++)
  - Submit by Friday 23:45
  - Discussions encouraged, but restrict those to concepts, not code.
- Exams
  - In-lab (2 hours 40 minutes)
- Your hackerrank id should be the same as your roll number.

Theory Exams

- Q1
  - 20 marks, 50 minutes
- Q2
  - 30 marks, ?
- EndSem
  - 50 marks, 3 hours
- All the exams would be open-book, open hand-written notes.
  - No electronics, printouts, photocopies.
  - Focus on concepts.

Programming and Data Structures

CS2700 and CS2710

Rupesh Nasre.
rupesh@iitm.ac.in

TAs: Maneesh, Sidharth, Deepak, Deepak,
Ananya, Brintha, Niket, Omji, Pranjal.

Course webpage: ~rupesh/teaching/pds/jul19
Moodle: https://courses.iitm.ac.in/course/view.php?id=5107

July 2019