Computational Models - Introduction[1]

Handout Mode

Instructors:
Prof. Ronitt Rubinfeld *rubinfeld at post.tau.ac.il*
Dr. Iftach Haitner *iftach.haitner at cs.tau.ac.il*

Teaching Assistants:
Dean Doron *deandoron at mail.tau.ac.il*
Yuval Moskovitch *mosyuval at gmail.com*
Oren Salzman *orenzalz at post.tau.ac.il*

Tel Aviv University. Spring Semester, 2016. Mondays, 13–16 and Wednesdays, 10–13.

February 29/ March 02, 2016

Part I

Administrativia

Administrativia

Course website: http://tau-cm2016b.wikidot.com

Site (containing a forum) is our sole mean of disseminating information.

Course Requirements:

- 6 problem sets (10% of grade).

  Submission via Moodle (see the course website)

  - Readable, concise, correct answers expected.
  - Late submission will not be accepted. (You have between one and two weeks, start working when you get them. Any excuse has to cover all the period.)
  - See more instructions on the course website.

  - Solving problems independently is highly recommended.
  - (variants of) questions from HW in exams!

AdministraTrivia II

- Midterm, covering first 6 lectures (up to computability). We will discuss it when we get closer.
- Midterm (magen) is scheduled to Friday, April 15, 2016.
- Final exam, covering all course material.
- You must pass the final exam to get a passing course grade.
- Final Grade: $0.75 \cdot \mathit{Exam} + 0.15 \cdot \max\{\mathit{Midterm}, \mathit{Exam}\} + 0.10 \cdot \mathit{HW}$.
- Prerequisites (formally): Extended introduction to computer science
  - But most importantly is “mathematical maturity”.
  - Students from other disciplines with mathematical background encouraged to contact the instructor.
- Textbook: Sipser — *Introduction to the theory of computation*, first or second editions.
- Other (excelent) book: Hopcroft, Motwani, and Ullman — *Introduction to Automata Theory, Languages, and Computation*.

Part II

Course overview

Why study theory?

- Basic computer science issues
  - What is a computation?
  - Are computers omnipotent?
  - What are the fundamental capabilities and limitations of computers?

- Pragmatic reasons
  - Avoid intractable or impossible problems.
  - Apply efficient algorithms when possible.
  - Learn to tell the difference.

Course Topics

- Automata Theory: Basic model of computation.
  - Re-invented in many other disciplines.
- Computability Theory: What can computers do?
  - True impossibility results.
- Complexity Theory: What makes some problems computationally hard and others easy?

Automata theory – Simple models

- Finite automata.
  - Related to controllers and hardware design.
  - Useful in text processing and finding patterns in strings.
  - Probabilistic (Markov) versions useful in modeling various natural phenomena (*e.g.* speech recognition).
- Push down automata.
  - Tightly related to a family of languages known as context free languages.
  - Play important role in compilers, design of programming languages, and studies of natural languages.

Computability Theory

In the first half of the 20th century, mathematicians such as Kurt Göedel, Alan Turing, and Alonzo Church discovered that some fundamental problems cannot be solved by computers.

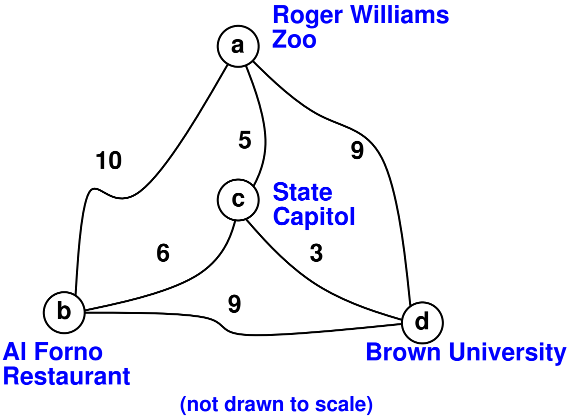
- Proof verification of statements can be automated.
- It is natural to expect that determining validity can also be done by a computer.
- Theorem: A computer cannot determine if mathematical statement true or false.
- Results needed theoretical models for computers.
- These theoretical models helped lead to the construction of real computers.

Complexity Theory

Key notion: tractable vs. intractable problems.

- A problem is a general computational question:
  - description of parameters
  - description of solution
- An algorithm is a step-by-step procedure
  - a recipe
  - a computer program
  - a mathematical object
- We want the most efficient algorithms
  - fastest (usually)
  - most economical with memory (sometimes)
  - expressed as a function of problem size

Example: Traveling Salesman Problem

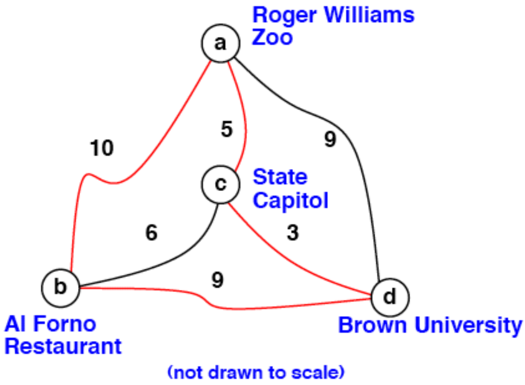


Input:

- set of cities
- set of inter-city distances

Goal: want the shortest tour through the cities

Example: Traveling Salesman Problem



Example: a, b, d, c, a has length 27

Part IV

Appendix
Not Taught in Class

Section 1

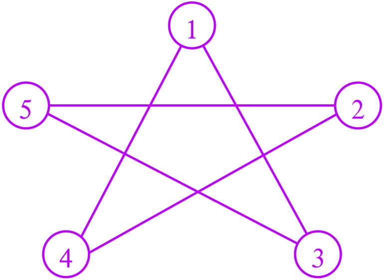
A (Very) Short Math Review

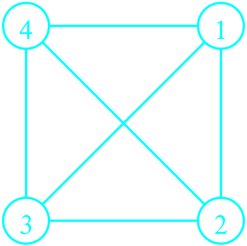
A Very Short Math Review

- Graphs
- Strings and languages
- Mathematical proofs
- Mathematical notations (sets, sequences, ...) $\surd$
- Functions and predicates $\surd$

$\surd$ = will be done in recitation.

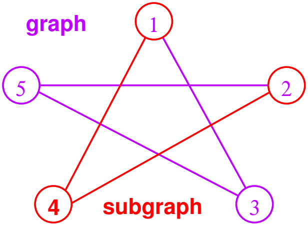
Graphs

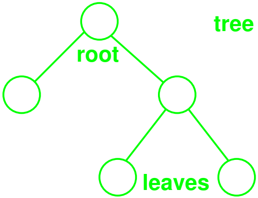


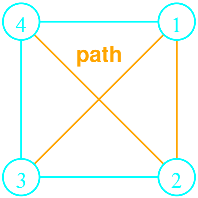


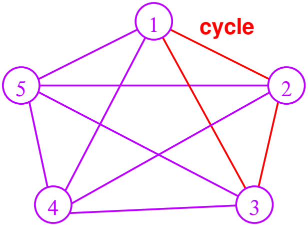
- $G = (V, E)$, where
  - V is set of nodes or vertices, and
  - E is set of edges
  - degree of a vertex is number of edges

Graphs

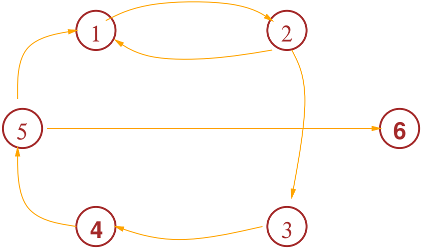

graph
1
2
3
4
5
subgraph

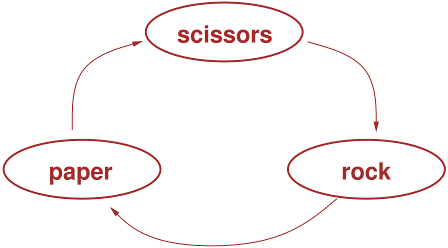

tree
root
leaves


4
1
path
3
2


1
cycle
5
2
4
3

Directed Graphs





Directed Graph and its Adjacency Matrix

Which directed graph is represented by the following 6-by-6 matrix?

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$A(i,j) = 1 \longleftrightarrow (i,j) \in E$

Strings and Languages

- an alphabet is a finite set of symbols
- a string over an alphabet is a finite sequence of symbols from that alphabet.
- the length of a string is the number of symbols
- the empty string ε
- reverse: *abcd* reversed is *dcba*.
- substring: *xyz* in *xyzzy*.
- concatenation of *xyz* and *zy* is *xyzzy*.
- x^k is $x \cdots x$, k times.
- a language $\mathcal{L}$ is a (possibly infinite) set of strings.

Proofs

We will use the following basic kinds of proofs.

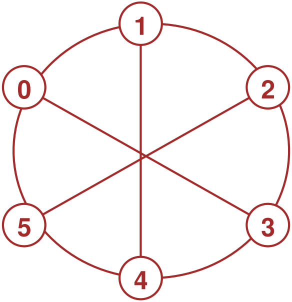
- by construction
- by contradiction
- by induction
- by reduction
- we will often mix them.

Proof by Construction

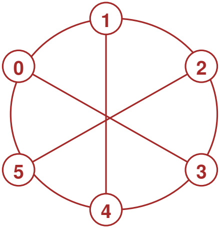
A graph is *k-regular* if every node has degree k.

Theorem 1

For every even $n > 2$, there exists a 3-regular graph with n nodes.



Proof by Construction



Proof: Construct $G = (V, E)$, where $V = \{0, 1, \ldots, n-1\}$ and

$$\begin{aligned} E &= \{\{i, i+1\}:\ \text{for } 0 \le i \le n-2\} \cup \{n-1, 0\} \\ &\quad \cup \{\{i, i+n/2\}:\ \text{for } 0 \le i \le n/2-1\}. \\ degree(i) &= |\{(i, i+1), (i, i-1), (i, i+n/2)\}| = 3 \end{aligned}$$

Missing details?

Note: A picture is helpful, but it is not a proof!

Proof by Contradiction

Theorem 2

$\sqrt{2}$ is irrational.

Proof: Suppose not, then $\sqrt{2} = \frac{m}{n}$, where m and n are relatively prime.

$$n\sqrt{2} = m \Longrightarrow 2n^2 = m^2$$

So m^2 is even $\implies$ m is even (?). Let $m = 2k$.

$$2n^2 = (2k)^2 = 4k^2$$
$$\Longrightarrow n^2 = 2k^2$$

Thus n^2 is even, and so is n. Therefore both m and n are even, and not relatively prime!

Proof by Induction

Prove properties of elements of an infinite set.

$$\mathbb{N} = \{1, 2, 3, \ldots\}$$

To prove that $\wp$ holds for each element, show:

- *base step:* show that $\wp(1)$ is true.
- induction step: show that if $\wp(i)$ is true (the induction hypothesis), then so is $\wp(i+1)$.

Induction Example

Theorem 3

All cows are the same color.

Base step: A single-cow set is definitely the same color.
Induction Step: Assume all sets of i cows are the same color.
Divide the set $\{1, \ldots, i+1\}$ into $U = \{1, \ldots, i\}$, and $V = \{2, \ldots, i+1\}$.

All cows in U are the same color by the induction hypothesis.
All cows in V are the same color by the induction hypothesis.
All cows in $U \cap V$ are the same color by the induction hypothesis.

Hence, all cows are the same color.

Quod Erat Demonstrandum (QED).

Induction Example (cont.)

(cows' images courtesy of www.crawforddirect.com/ cows.htm)

Proof by Reduction

We can sometime solve problem A by reducing it to problem B, whose solution we already know.

Example: k^{th} element using MAX element

Input: $S = \{7, 12, 3, 15, 9, 5, 19\}$, $k = 2$

Reduction to MAX

- For $i = 1, \ldots, k-1$ Do:
- $M = MAX(S)$
- $S = S \setminus \{M\}$
- Next i
- Output $MAX(S)$