Review/study guide (caution: not exhaustive!)

COS116: 5/6/11

Sanjeev Arora

Scribbler

Sensors, commands,
understanding scribbler pseudocode

Algorithms

- What they are.
- Algorithms like findmin, bubblesort, binary search, simulation of game of life.
- Running times of above algorithms, and ability to estimate running time of simple pseudocode.
- Conceptual understanding of other algorithms encountered: pagerank, k-means, stable matching etc.

Computer music

- Digital representations and what they mean (bits, bytes, sampling rates etc.)
- Conceptual understanding of spectral representation.
- Understanding of music synthesis and related notions.

Undecidability etc.

- What it means. Salient implications.
- Turing post programs.
- Understanding of the proof by contradiction idea and how it is used to prove undecidability of halting problem.
- Examples of other undecidable problems.
- Understanding of simple self-reproducing program.

Boolean logic

- Representations: formula, circuit diagram, truth table. How to convert between them.
- Combinational circuits vs sequential circuits. How “memory” is implemented using feedback.
- Flip flop, and its use in clocked circuits
- Timing diagrams
- Designing FSMs and synchronous circuits

Computer architecture

- Some idea of the kinds of instructions used in machine language, and the fetch-decode-execute cycle used to implement it

Memory hierarchies

- Caching: what, why, how.
- Ability to compute performance of simple memory architectures.
- Replacement schemes (eg LRU)
- Multitasking: what, why, how.
- Scheduling: considerations and ideas (round robin, priority based)

Computer security

- Ways of “breaking in”
- Viruses/worms. How they spread. (Quantitative understanding based upon lab.)
- Zombies/botnets. How botnets are created. How researchers study them.
- Familiarity with other issues in security.

P vs NP

- “Needle in haystack” problems (NP)
- Meaning of P, NP, NP-complete, and examples of problems of each type.
- Meaning of “exponential time”/exhaustive search.

Cryptography

- Simple letter-scrambling cyphers
- One-time pad (modern version); its strength and limitations.
- What does it mean to “not learn anything”.
- Public-key crypto; conceptual understanding (box lockable by all; openable only by owner).
- Zero knowledge proofs.

Graphics/vision

- Meaning of pixel, color representation, brightness/contrast
- Computer vision: edge detection (simple algorithm). Some understanding of higher-level tasks.
- Graphics: triangulated representations, different ideas for rendering, reflectance models (types of reflection).

Machine learning

- What it is: learning from data/experience
- Simple example: clustering.
- K-means algorithm
- Spam detection algorithm

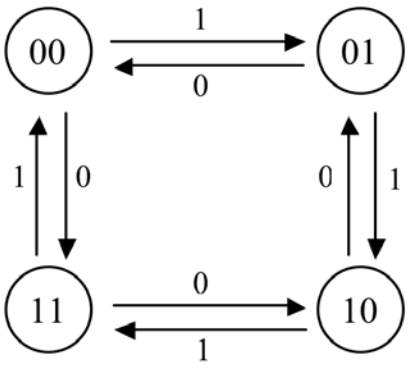
Artifical Intelligence

- Psychological explanations of intelligence: behaviorism vs inbuilt structures
- Neural networks (Mccullough-Pitts model)
- Simulation argument for AI
- Turing test: meaning, strengths/weaknesses, Searle's objection, etc.

Last lecture

- Conceptual understanding of economic bubbles and the connection to reasoning about distributed systems.
- Conceptual understanding of other topics discussed.

1. (30 points) The diagram below depicts a finite state machine (FSM).



(a) How many states does this machine have? How many bits does the input have?

(b) Suppose we fix the input of the machine to 0. Describe in a line what the machine does.

(c) Write the truth table for this finite state machine. The table should specify the next state for each possible combination of input and current state. (10 point)

What does this pseudocode do? What is the running time in terms of n?

```
Get n, A[1], …, A[n];
result ← 0;
i ← n;
c ← 1;
Do while (i > 0)
{
 result ← result + A[i] * c;
 i ← i-1 ;
 c ← c*2;
}
Print "The result is: "
Print result
```