CSE 113 B

April 19-23, 2010

Announcements

- Exam 4 Review – Wednesday, April 21st
- Exam 4 – Friday, April 23rd
- Lab 4 due - Sunday, April 25th
- Exam return – Monday, April 26th

3

Today's Lecture

- There are problems we can solve
- There are problems we know we can solve, but they are expensive to solve
- There are problems we know we can not solve

4

Problems we can solve

- You've solved a few yourself.
- There are obviously many more.
- What kinds of problems have computers solved?

5

PROBLEMS WE KNOW WE CAN SOLVE

- These problems have solutions that run in reasonable time
  - When we discuss this formally, we express the time it takes to find a solution to be a function where the variable is the size of the inputs. The function is expressed in terms of the size of the inputs.
  - For example, a reasonable solution may be expressed in terms n^3, so if n = 100, then when we cube it, we get 100,000

6

UNREASONABLE TIME

- When we come across a problem that runs in unreasonable time, we see that for 100 inputs, the resulting function returns a value of 1.27×10^{30}
- Which is the number 1,270,000,000,000,000,000,000,000,000,000
- If the computer uses 1 second to process an input, that would be 21,167,000,000,000,000,000,000,000,000 minutes, or 352,778,000,000,000,000,000,000,000 hours, or 1,469,910,000,000,000,000,000,000 days or 40,271,400,000,000,000,000,000 years

7

A SIDE NOTE

- 1.27 x 10^{30} is just shy of the estimate of the total number of atoms in the observable universe, which is guessed to be about 4 x 10^{81}
- These are the values if the function is 2^n where n is the number of inputs.
- If we use another function n!, the result when n = 100 is 9.3 x 10^{157} (larger than the estimate of the total number of atoms in the observable universe).

8

UNREASONABLE TIME

- Just because something runs in unreasonable time doesn't mean we don't keep trying to solve it in reasonable time.
- Sometimes, we can improve the way we think about things and get to solutions that are reasonable.

9

A "Hard" Problem

- Example: Computer player for chess. The estimate for the total number of board configurations for chess is somewhere in between the values we mentioned on the previous slide (10^{50}).
- Therefore, for the computer player to know how to win, it needs to know each of those board configurations and how to win if the board is in that configuration.
- Well, sort of – there are shortcuts to this, which is how we got a computer that is able to play chess.

10

Unreasonable Problems

- Putting together an n-piece jigsaw puzzle
- "Tetris" – not quite Tetris, but a similar class of problems using Tetris – like pieces
- Traveling Salesman
- Scheduling problems (N teachers, M hours, P classes) – schedule so that all P classes are covered so that no teacher is teaching at the same time two different classes, nor that the same class is being taught at the same time by two different instructors

11

Unreasonable Problems

- We know that there are unreasonably-timed solutions to these problems in the general case, but for some if the variables (M, N, etc) are small, we can come up with reasonably-timed solutions.
- Also, for these problems, you can check to see if a proposed solution is in fact valid in reasonable time
- These problems belong to a specific class of problems with these characteristics.

12

Some unanswered questions

- A currently un-answered (and potentially never-answered) question is whether or not there are reasonably-timed solutions to our currently known unreasonably-timed solution problems. We know that sometimes we can find reasonable solutions, but can we find general-case reasonably-timed solutions?

13

Two more problems

- If given a description of a problem to solve and a solution to the problem designed by a student – can we write a program to verify that the program solves the problem?
- Can we write a program to verify that there are no infinite loops in a program?

14

Answers…

- In fact, the answer is no to both of these questions.
- Furthermore, it's not just a "I don't think so", it's a provable fact that we will never be able to solve these problems.
- These problems exist in a class of problems known as "undecideable" problems – we know that we can never solve these problems using a computer.