CompSci 101
Introduction to Computer Science

December 4, 2014

Prof. Rodger

Evaluation

- Fill out course evaluation on ACES
- On Sakai (under announcements) please rate your Lab UTAs and any other UTAs you interacted with

Announcements

- Submit works now!
- Final Exam – accommodations/reschedule?
  - Fill out form by Dec 5
- APT 10 due Friday, last late day is Dec 7
- Asg 8 due tonight!, Last late day is Dec 6
- Asg 9 due Friday night, not accepted after midnight!
- Today
  - More on sorting, Classwork, CS story

More Announcements

- Be a UTA
  - http://www.cs.duke.edu/csed/uta/
- Next course
  - CompSci 201
  - Start all over again with Java
  - Java has if, loops, lists, maps (dictionaries), sets
  - Is that familiar?
  - Learn about nonlinear structures that can be more efficient

Final Exam

- Sec 01 (White Lect. Hall) – Sat Dec 13 2pm
- Sec 02 (LSRC B101) – Wed Dec 10 7pm
- Closed Book, Closed Notes, Closed neighbor
- Python Reference Sheet
- Covers all topics through today
- Best way to study is practice writing code!
- See old tests (no old final exams)
- Fall 2014 tests – see other section tests

Final Exam (cont)

- Test format
  - Multiple choice
  - Writing code
- Topics include:
  - if, loops, lists, sets, maps, files, functions
  - recursion and regular expressions – reading level only

Calculate Your Grade

- From “About” tab on course web page

labs	10%
quizzes(reading or knowledge)/classwork	10%
apts	10%
assignments	20%
two exams	25%
final exam	25%

More on Grades

- Lecture – drop the first two weeks (drop/add period) plus 3 more
- Reading Quizzes – will drop 20 points
- Lab – drop 8 points (each lab is 4 pts)

Wrap up Sorting

- Different ways to sort?
  - Over 50 sorting algorithms
- What sorting algorithm does Python sort use?
- Does President Obama know his sorts?
- Sorting animations

  http://www.sorting-algorithms.com/

Merge Sort

- Idea: Divide and Conquer
- Divide array into two halves
- Sort both halves (smaller problem)
- Merge the two sorted halves

- Learn about this and other sorts in CompSci 201, also how to analyze them to determine which one works best.
- Timsort
- Shellsort

Classwork

bit.ly/101fall14-1204-01

Growth of functions

- As the size of the data increases, how many steps are there for an algorithm/method?

Timings

N	$\log_2 N$	N^2	N^3	2^N
10	3.3	100	1000	1024
20	4.3	400	8000	1048576
40	5.3	1600	64000	1.1×10^{12}
80	6.3	6400	512000	1.2×10^{24}
160	7.3	25600	4096000	1.4×10^{48}

Timings

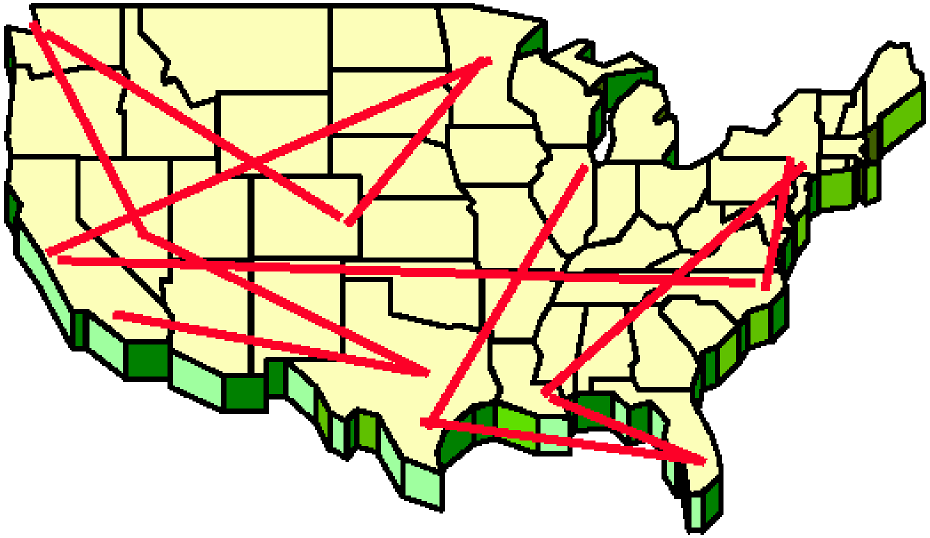
N	$\log_2 N$	N^2	N^3	2^N
250	7.9	62,500	1.56×10^7	1.8×10^{75}
500	8.9	250,000	1.25×10^8	3.2×10^{150}
1000	9.9	1×10^6	1×10^9	
2000	10.9	4×10^6	4×10^6	
4000	11.9	1.6×10^7	8×10^9	

Look at the timings of the sorts

- How do the sorts compare?
  - With size as they grow
  - With different types of data
    - Random
    - Reverse
    - Almost sorted

Problem: Traveling Band

- Band wants you to schedule their concerts.
- They don’t like to travel. Minimize the time they are on the bus!
- Given N cities, what is the best schedule (shortest distance) to visit all N cities once?



How do you calculate the best path?

- Try all paths
  - Atlanta, Raleigh, Dallas, Reno, Chicago
  - Dallas, Atlanta, Raleigh, Reno, Chicago
  - Etc.
- Would you agree to code this up?

How long?

Number of Cities	All paths – N!	Time to solve - 10^9 Instructions per second
10	3 million	
15	10^{12}	
18	10^{15}	
20	10^{18}	
25	10^{25}	

How long?

Number of Cities	All paths – N!	Time to solve - 10^9 Instructions per second
10	3 million	< sec
15	10^{12}	
18	10^{15}	
20	10^{18}	
25	10^{25}	

How long?

Number of Cities	All paths – N!	Time to solve - 10^9 Instructions per second
10	3 million	< sec
15	10^{12}	16 min
18	10^{15}	
20	10^{18}	
25	10^{25}	

How long?

Number of Cities	All paths – N!	Time to solve - 10^9 Instructions per second
10	3 million	< sec
15	10^{12}	16 min
18	10^{15}	11 days
20	10^{18}	
25	10^{25}	

How long?

Number of Cities	All paths – N!	Time to solve - 10^9 Instructions per second
10	3 million	< sec
15	10^{12}	16 min
18	10^{15}	11 days
20	10^{18}	31 years
25	10^{25}	

How long?

Number of Cities	**All paths – N!**	**Time to solve - 10^9 Instructions per second**
10	3 million	< sec
15	10^{12}	16 min
18	10^{15}	11 days
20	10^{18}	31 years
25	10^{25}	10^8 years

$P = NP$?

- P: Problems with polynomial time solutions
  - N, N^2
  - Example: Selection sort
  - Easy to solve
- NP: problems with not polynomial time solutions
  - 2^n , $N!$
  - Hard to solve

Does P = NP?

- Famous CS question
- If yes, a whole class of difficult problems can be solve efficiently, one problem is reducible to another
- If no, none of the hard problems can be solved efficiently

A CS Story