CS 4104: Data and Algorithm Analysis

August 26, 2014
Course Information

- Instructor
  - Sharath Raghvendra, 2160H Torgerson, 231-6256, sharathr@cs.vt.edu
  - Office Hours: 2:30pm–4:30pm, Wednesday and by appointment

- Teaching assistants
  - Sorour Ekhtiar Amiri (Graduate TA), esorour@vt.edu
  - Office Hours: To be determined
  - Gustavo Arango Argoty (Graduate TA), gustavo1@vt.edu
  - Office Hours: To be determined
Course Information

- **Instructor**
  - Sharath Raghvendra, 2160H Torgerson, 231-6256, sharathr@cs.vt.edu
  - Office Hours: 2:30pm–4:30pm, Wednesday and by appointment

- **Teaching assistants**
  - Sorour Ekhtiari Amiri (Graduate TA), esorour@vt.edu
  - Office Hours: To be determined
  - Gustavo Arango Argoty (Graduate TA), gustavo1@vt.edu
  - Office Hours: To be determined

- **Class meeting time**
  - Tuesday and Thursdays 2:00pm–3:15pm, Lavery 320
Course Information

► Instructor
  ► Sharath Raghvendra, 2160H Torgerson, 231-6256, sharathr@cs.vt.edu
  ► Office Hours: 2:30pm–4:30pm, Wednesday and by appointment

► Teaching assistants
  ► Sorour Ekhtiari Amiri (Graduate TA), esorour@vt.edu
  ► Office Hours: To be determined
  ► Gustavo Arango Argoty (Graduate TA), gustavo1@vt.edu
  ► Office Hours: To be determined

► Class meeting time
  ► Tuesday and Thursdays 2:00pm–3:15pm, Lavery 320

► Prerequisite: Grade of C or better in CS 3114; P or better in MATH 3034 or MATH 3134
Keeping in Touch

- Course web site
  http://courses.cs.vt.edu/~cs4104/sharath/fall2014,
  updated regularly through the semester
- Scholar web site: grades, slides and homework/exam solutions
- Scholar mailing list: announcements
- Piazza sign-up link: Signup Link
  piazza.com/vt/fall2014/cs4104
Required Course Textbook

- Algorithm Design
- Jon Kleinberg and Éva Tardos
- Addison-Wesley
- 2006
Reading Assignments

- Reading assignment available on the website.
- Read **before** class.
Lecture Slides

- Will be available on course website.
- Usually posted before class.
- Class attendance is extremely important.
Lecture Slides

- Will be available on course website.
- Usually posted before class.
- **Class attendance is extremely important.** Slides may not be available for a few lecture classes.
Homeworks

- Posted on the web site \(\approx\) one week before due date.
- Announced on the class mailing list.
Homeworks

- Posted on the web site \( \approx \) one week before due date.
- Announced on the class mailing list.
- Collaboration allowed but prepare solutions individually and acknowledge those who you collaborated with.
- Hand in hard-copy of the solution.
Homeworks

- Posted on the web site ≈ one week before due date.
- Announced on the class mailing list.
- Collaboration allowed but prepare solutions individually and acknowledge those who you collaborated with.
- Hand in hard-copy of the solution.
  - Hand-written solutions must be legible, otherwise no credit will be awarded.
Homeworks

- Posted on the web site ≈ one week before due date.
- Announced on the class mailing list.
- Collaboration allowed but prepare solutions individually and acknowledge those who you collaborated with.
- Hand in hard-copy of the solution.
  - Hand-written solutions must be legible, otherwise no credit will be awarded.
- Homework grading policy: strict for easy questions, lenient for the "hard" questions.
Examinations

- Take-home midterm.
- Take-home final (comprehensive).
- 1 week deadline from the time of release.
- No collaborations on exams, submit hard-copy of the solutions.
Grades

- Homeworks: $\approx 6-8$, 60% of the grade.
- Take-home midterm: 15% of the grade.
- Take-home final: 25% of the grade.
Funding

- Some Undergraduate Research Assistant positions available for summer.
- Please talk to me if interested.
Course Goals

- Model "real-world" problems mathematically
- Paradigms and principles to design efficient algorithms.
Course Goals

- Model "real-world" problems mathematically
- Paradigms and principles to design efficient algorithms.
- Learn techniques to prove correctness, efficiency and lower bounds for algorithms.
Topics

Course roughly follows topics suggested in textbook

- Measures of algorithm complexity ($O(\cdot)$, $\Omega(\cdot)$ etc.)
- An Example Problem (Stable Marriage Problem)
- Graphs (BFS, DFS with applications)
- Greedy algorithms
- Divide and conquer
- Dynamic programming
- Network flow problems (Bipartite Matching, Max-flow etc.)
- Linear Programming (Duality, Hungarian algorithm)
- NP-completeness (Concept of reductions)
- Approximation algorithms
- Streaming algorithms
What is an Algorithm?
What is an Algorithm?

Chamber’s  A set of prescribed computational procedures for solving a problem; a step-by-step method for solving a problem.

Knuth, TAOCP  An algorithm is a finite, definite, effective procedure, with some input and some output.