Teaching Staff

- Instructor: Ahmed Gheith
  - Day job: Research Staff @ IBM Austin Research Lab
  - class web site: www.cs.utexas.edu/~gheith/cs352h (goes live 9/1/2012)
  - I’ll be using blackboard to post and e-mail all official class related announcements
  - Office hours: T Th 5pm until last student leaves (or by appointment)
  - email: ag at gheith dot com (checked regularly, no grade or class performance discussions)
  - ag at utexas dot edu (checked on demand, never leaves the university systems)
References


- You will be writing some Scala programs. Online tutorials at [http://www.scala-lang.org/node/1305](http://www.scala-lang.org/node/1305)

- You will be using GIT to manage and submit your assignments.
  - type “man gittutorial” at Unix prompt
  - google “git tutorial”
  - Some help will be given in class

- Basic Unix command line skills and use of a text editor (vi, emacs, ...).

- Presentation slides will be used as needed and will be made available to you.
Lectures, assignments, etc

• TTh 3:30..5:00pm RLM 5.124: lectures by Ahmed and 1-2 guest lectures (time permitting)

• Reading assignments will be given. Please come prepared and ask as many questions as you want during the lecture. I will let you know if I feel the questions are not relevant to the rest of the students and I’ll be glad to discuss them after the lecture or during office hours

• No surprises (no pop quizzes). All quizzes will be pre-announced.

• Attendance is not taken but you have no way of doing well in the class without attending and taking good notes.

• Powerpoint presentations put me to sleep. There will be very few of them

• Programming assignments are almost always harder and take longer to finish than you originally anticipated.

• All assignments are due by 11:59pm on the due date. No exceptions without a university recognized emergency.

• Commit and push to GIT often. It’s to your advantage.
Grading

• Class participation: 10%
  • Sharing notes, asking good questions, participating in discussions
• Exams: 30% (final and 2 midterms)
  • [10%] Midterm #1
  • [10%] Midterm #2
  • [10%] Final
• Programming assignments: 40%
• Quiz / homework: 20%
• Optional exam/project: up to 10%
• Grade cutoff: A (>= 90%), B (>= 80%), C (>= 70%), D (>= 60%)
Class outline

• Review: computer architecture, instructions sets, logic design
• Single-cycle processor: design and evaluation
• Multi-cycle processor: design and evaluation
• Cache: design and evaluation
• Pipelining: design and evaluation
• Instruction-level parallelism: design and evaluation
• Concurrency and parallelism
How to do well in this class?

• You’re expected to have the basics covered:
  • binary encoding of data and instructions
  • combinational and sequential logic
  • basic programming skills
  • ability to learn new programming languages with little help

• Attendance and note-taking / sharing is crucial
  • we will spend lots of time using the black/white/green board
  • we will iterate over ideas and design and make mistakes and fix them together
  • you will be given links to online material but very little notes

• Ask lots of questions and participate in discussions. Help and ask for help.

• Most lectures will be conducted as design sessions at work. Feel free to think of me as your senior colleague. You’re encouraged to criticize my design decisions and come up with better proposals. I’ll leave some Easter eggs in the design for you to find.

• Start your assignments early and finish them on time