Administration

- **Course CSE2312**
  - What: Computer Organization & Assembly Language Programming
  - When: Mon. & Wed. 4:00 ~ 5:20pm
  - Where: PKH 111
  - Who: Song Jiang (song.jiang@uta.edu)
  - Office Hour: Mon. & Wed. 10:00 ~ 11:00pm at ERB 101
  - or by appointments
  - Homepage: http://ranger.uta.edu/~sjiang/CSE2312-fall-18/CSE2312.htm (Please visit this website regularly)

- **About your instructor**
  - Research areas: file and storage system, operating system, parallel and distributed computing, and high performance computing,

- **GTA**
  - Xingsheng Zhao (Office ERB 101), xingsheng.zhao@mavs.uta.edu (subject to change)
  - Office hours: TBA
Study Materials

- **Prerequisites**
  - Intermediate Programming (CSE 1320)
  - Introduction to Computers & Programming (CSE 1310)
  - What this really means:
    - You know at least one programming language.
    - You generally know how to write/run/test programs.
    - Elementary knowledge of math and algorithms

- **Preferred**
  - CSE2441 Intro to Digital Logic

- **Text book**
Grading

• Distribution
  – 20%  Homework Assignments
  – 20%  Programming Assignments
  – 10%  Quizzes
  – 20%  Midterm Exam
  – 30%  Final Exam

100%

• Attention
  – Homework is as important as any other aspects of your grade!
  – Attendance is strongly encouraged.
  – The university makeup policy will be strictly adhered to. Generally, no make-up exams/quizzes except for university sanctioned reasons.
  – When missing an exam/quiz due to unavoidable circumstances, PLEASE notify the instructor and request a makeup approval ahead of time.
Final Grade

• Final Letter Grade
  – [90 100] --- A
  – [80 90) --- B
  – [70 80) --- C
  – [60 70) --- D
  – [00 60) --- F

• Note
  – [ ] denotes inclusion and ( ) denotes exclusion.
  – Your final weighted scores may be curved before assignment of your letter grade.
Assignments

• **Homework assignments**
  – Posted on the website, typically due in one week at the beginning of the class
  – Automatic 20% deduction for each day late for up to two days
  – Submissions will not be accepted if overdue for three or more days

• **Programming assignments**
  – Posted on the website, typically due in 1~2 weeks
  – They will be submitted on the blackboard.

• **Collaboration**
  – You may discuss assignments with others, but must write up them on your own.
  – Failure to comply with this policy is a violation of academic integrity

• **Start early!**
Information

• **Course Webpage**
  – Visit the webpage regularly.
  – Announcements, assignments, and lecture notes will be posted there.

• **Grade Appeal**
  – You may appeal the grade in writing (email) within five days.
  – Appealed to the GTA firstly (if graded by GTA), then to the instructor, if necessary.
  – Please refer to the UTA Catalog for the detailed guide of grade appeals.

• **Drop Policy**
  – The university withdrawal policy will be strictly adhered to.

• **Others**
  – Accommodating students with disabilities
  – Student Support Services
Course Overview

• What is it?
  – Computer organization
  – Assembly language programming

• Why is the course about?
  – Hardware & Software

• Will I really use this knowledge and skills?
  – Necessary knowledge for any CS students
  – You may not become a professional assembly programmer or processor designer but you need to know how to use it and how it works

• How to succeed in this course?
Why Are You in this Class?

• Curiosity on how computer actually works.
• To obtain background and be prepared for more advanced CSE classes
• Understand the effect of the architecture on your code
  – Learn how to directly control the computer
• To have the necessary background to understand innovations in the hardware and software
  – Your desktop, laptop, ipad, iphone, etc.
• Not necessary to become a professional assembly programmer
  – But to be able to write, debug, and tune small-scale codes.
  – To be able to understand innovative ideas.
Why Assembly?

• **Two main reasons**
  – Accessibility to system hardware
  – Space and time efficiency

• **Accessibility to system hardware**
  – Assembly Language is useful for implementing system software
  – Also useful for small embedded system

• **Space and time efficiency**
  – Understanding sources of program inefficiency
  – Tuning program performance
  – Writing compact code
What will You Get Out of the Course?

- **Learn basic elements of computer systems**
  - CPU, memory, and I/O.
- **Understand how various sub systems function in one system.**
- **Design issues**
  - Performance vs cost tradeoffs
- **Interaction of software (programs) and the underlying hardware (on which programs execute)**
Course Goals

• **Teaching Style**
  – Architecture course have been traditionally taught in two way: top-down or bottom-up.
  – We’re going in from the middle. programmer-centric approach.

• **Teaching Goals**
  – Show that by knowing more about the underlying system, one can be more effective as a programmer.
  – Write programs that are more reliable and efficient.
  – Understand how programs interact with the underlying hardware.
  – Learn the internal and external of a computer’s architecture.
  – Learn how to do low-level programming.
Course Expectation

• **What to expect from the course:**
  – Will cover key issues and concepts in class.
  – Programming projects
  – A midterm exam and a final exam
  – Homework and quizzes
  – Will provide review and teach you the tools used in the projects.

• **What do I expect of you:**
  – Attend classes
  – Read the textbook in addition to having lectures
  – Work through the problems in the textbook (not really homework… but it helps)
  – Do projects
  – Ask questions (IMPORTANT)