Course information

- Welcome to CS/CoE0447!
- See separate “Course Outline”
- Course web page:
  - http://www.cs.pitt.edu/~cho/cs0447
- Instructor:
  - Dr. Sangyeun Cho (cho@cs.pitt.edu, 383-7018)
  - Office hours: M/Th 2pm – 4pm, @5407 SENSQ
- TA:
  - Kyeon Lee (lee@cs.pitt.edu)
  - Office hours: M 4pm – 6pm, W 2pm – 3pm @5802 SENSQ

Final grade

- Based on your obtained points
- “The” table

<table>
<thead>
<tr>
<th>Grade</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>280 – 300</td>
</tr>
<tr>
<td>B+</td>
<td>250 – 259</td>
</tr>
<tr>
<td>C+</td>
<td>220 – 229</td>
</tr>
<tr>
<td>D+</td>
<td>190 – 199</td>
</tr>
<tr>
<td>A</td>
<td>270 – 279</td>
</tr>
<tr>
<td>B</td>
<td>240 – 249</td>
</tr>
<tr>
<td>C</td>
<td>210 – 219</td>
</tr>
<tr>
<td>D</td>
<td>180 – 189</td>
</tr>
<tr>
<td>A-</td>
<td>260 – 269</td>
</tr>
<tr>
<td>B-</td>
<td>230 – 239</td>
</tr>
<tr>
<td>C-</td>
<td>200 – 209</td>
</tr>
<tr>
<td>D-</td>
<td>170 – 179</td>
</tr>
</tbody>
</table>

- There is no curve

Course information


Student evaluation

- 11 lab. assignments (LA): 110 points total out of 300
- 4 homework assignments (HA): 60 points total
- 3 programming assignments (PA): 60 points total
- 2 mid-term exams: 40 points total
- 1 final exam: 30 points
- GRAND TOTAL = 300 (the maximum point you can get)
Course policies

- Exams
  - Closed book; calculators are OK
  - No make-up exam will be given unless there is a valid and approved excuse (in advance)
  - Result of cheating
    - 1st time: 0 points
    - 2nd time: automatic F grade
- Labs.
  - You must come to the labs (attendance accounts for 5 points out of 10 per lab)
  - You will be given a lab assignment each week
  - Each assignment can be done during the lab; submission deadline is the next lab meeting in a week
  - You will collaborate with a partner during the lab; however, each will submit the completed work individually
- Homework assignment (HA) and programming assignments (PA)
  - Assignments must be done alone
  - Assignments done in collaboration, if detected, will be given “0” points
  - Late submissions are not accepted

Important dates

- 9/5: ADD/DROP deadline
- 9/16: mid-term exam #1 (preparatory mock exam 9/11)
- 10/21: mid-term exam #2 (mock exam 10/16)
- 10/14: no classes due to Fall Break
- 11/25,27: no class due to Thanksgiving
- 12/10: Final exam (mock exam 12/2)

Computer systems

- Why do we call a computer a computer?
- What makes a computer a computer?
- “Desktop computers”
  - Examples include PC, Mac, …
  - Notebooks

Desktop computers
Computer systems

- Why do we call a computer a computer?
- What makes a computer a computer?

- ”Desktop computers”
  - Examples include PC, Mac, …
  - Notebooks

- ”Servers”
  - Web servers
  - Supercomputers

- ”Embedded computers”
  - Hidden inside something not computer
  - Applications that run on these computers are specific

Servers

Embedded computers
Common factors

- There may be different forms of “computation”
  - Example: digital TV tuner that converts a compressed digital motion picture format into something that we can view

- We are interested in a *programmable computing machine* or a *processor*
  - Desktop computers
  - Servers
  - Embedded computers

- What are the common factors that make a computer a computer?

In CS/CoE0447

- We study
  - Computer architecture
    - MIPS as the example architecture
  - Basic concepts of system software such as assembler, linker, compiler
  - Basic computer arithmetic
    - Binary numbers
    - Operations (add, sub, …)
  - Basic logic design
  - Basic processor performance analysis
  - Processor organization
    - Datapath
    - Control

- We do
  - Assembly language programming (using MARS simulator)
  - MARS (written by a Pitt graduate!) [http://www.cs.missouristate.edu/MARS/](http://www.cs.missouristate.edu/MARS/)
### Computer architecture?

- It’s not about doing “architecture” with a computer.
- It’s about designing a computer system (esp. hardware).

### Computer architecture?

- We are interested in **principles** in designing computer hardware in this course and programming it at the lowest level.
- Computer systems
  - Underlying hardware
  - Software running on it
- Computer architecture
  - The hardware/software interface seen by the user (as a programmer)
  - Instruction set architecture (ISA)
- Processor microarchitecture
  - Implementation of a given architecture
  - May or may not be visible to the user

### Layers or views

- Our view of a computer system in this course is centered around the interface between the lowest level in software and the hardware.
- We will talk a lot about assembly or machine **instructions**.
- It’s like learning a whole new language!