Welcome to CS 340: Introduction to Computer Systems
Course Website: https://courses.grainger.illinois.edu/cs340/

Description: Basics of computer systems. Number representations, assembly/machine language, abstract models of processors (fetch/execute, memory hierarchy), processes/process control, simple memory management, file I/O and directories, network programming, usage of cloud services. 3 credit hours.

Staff:
Prof. G Carl Evans <gcevans@>, Course Instructor
Teaching Associate Professor of Computer Science, Grainger College of Engineering

TAs: Ramya Bygari (rbygari2)
Evan Matthews (evanmm3)

Coursework and Grading
A total of 1,000 points are available in CS 340, along with many opportunities to earn extra credit. The points are broken down in the following way:

- **140 points:** Homeworks (1-2 /week)
  - Points even divided between the homeworks
  - Usually on PrairieLearn, but occasionally another platform

- **200 points:** Midterm Exams in CBTF (2 × 100 points)
  - Midterm 1 Exam (CBTF): Thurs, March 2 - Sat, March 4
  - Midterm 2 Exam (CBTF): Thurs, April 27 - Sat, April 29

- **440 points:** Machine Projects (11 weeks × 40 points)
  - Weekly machine problems, released every Tuesday and due the following Tuesday with a Wednesday grace period.
  - Extra credit for completing early milestones and completion.

- **220 points:** Final Project
  - Multi-week Final Project, presented during the final exam period instead of a final exam (no final exam!)
  - **Must be present on Tuesday, December 12, 2023.**

We never curve individual exam or assignment scores. Instead, if necessary, we may lower the points required for each grade cutoff to be lower than the stated cutoff. In no case will we raise the stated cutoff, so having 930 points will always earn you an “A” in the course.

---

**Final Course Grades**
Your course grade is determined by the number of points you earn:

<table>
<thead>
<tr>
<th>Points</th>
<th>Grade</th>
<th>Points</th>
<th>Grade</th>
<th>Points</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exceptional</td>
<td>A+</td>
<td>[930, 1070)</td>
<td>A</td>
<td>[900, 930)</td>
<td>A-</td>
</tr>
<tr>
<td>[870, 900)</td>
<td>B+</td>
<td>[830, 870)</td>
<td>B</td>
<td>[800, 830)</td>
<td>B-</td>
</tr>
<tr>
<td>[770, 800)</td>
<td>C+</td>
<td>[730, 770)</td>
<td>C</td>
<td>[700, 730)</td>
<td>C-</td>
</tr>
<tr>
<td>[670, 700)</td>
<td>D+</td>
<td>[630, 670)</td>
<td>D</td>
<td>[600, 630)</td>
<td>D-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(600, 0)</td>
<td>F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Foundations of Computer Systems**
There are six major components to a computer, which we will refer to as the “foundations” of a computer system:

1. Simulation of machines
2. Programs
3. Operating systems
4. Networks
5. Databases
6. Cloud services

**System-level Abstractions**
After covering the “foundations”, we will begin to abstract the entire system as single node and explore more complex topics:

1. Asynchronous communication
2. Distributed systems
3. Cloud computing
Representing Data: Binary
All data within a computer is ____________; either 0 or 1.

Converting between base-2 and base-10:

\[ \begin{align*}
1_2 &= 10 \\
10_2 &= 10 \\
11_2 &= 10 \\
100_2 &= 10 \\
\end{align*} \]

Representing Data: Hexadecimal
Binary data gets really long, really fast! The number of students enrolled at University of Illinois is \texttt{0b1100 1100 0110 1011}
- To represent binary data in a compact way, we often will use hexadecimal -- or “base-16” -- denoted by the prefix \texttt{0x}.

Hexadecimal Digits:

<table>
<thead>
<tr>
<th>Place of Hexadecimal Numbers:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex Number:</td>
</tr>
<tr>
<td>(x) Place Value:</td>
</tr>
<tr>
<td>Decimal Place Value:</td>
</tr>
<tr>
<td>SUM:</td>
</tr>
</tbody>
</table>

Translation from Decimal to Hexadecimal:

\[ \begin{align*}
11_{10} &= \texttt{0x} \\
34_{10} &= \texttt{0x} \\
87_{10} &= \texttt{0x} \\
255_{10} &= \texttt{0x} \\
\end{align*} \]

Hexadecimal is particularly useful as it ________________:

University of Illinois student population last Fall (52,331):

<table>
<thead>
<tr>
<th>Student Population:</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{0b}</td>
</tr>
<tr>
<td>\texttt{0x}</td>
</tr>
</tbody>
</table>

Number of people following Taylor Swift on Instagram (240,825,376):

<table>
<thead>
<tr>
<th>Following Number:</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{0b}</td>
</tr>
<tr>
<td>\texttt{0x}</td>
</tr>
</tbody>
</table>

\texttt{01/binary.c}

```c
4  int v1 = \texttt{0b10010};
5  int v2 = \texttt{0b11001};
6  int v3 = v1 + v2;
7  printf("%d\n", v3);
```

\texttt{01/hex.c}

```c
4  int h1 = \texttt{0xc0ffee};
5  int h2 = \texttt{0xf00d};
6  printf("%x\n", h1 + h2);
```