CS 216
Introduction to Software Engineering
Logistics
david.b.brown@uky.edu
35% - Programming
30% - Weekly Assignments
25% - Tests
  10% - Midterm
  15% - Final
10% - Attendance
Typical week:
Practicum assignment due Thursday at midnight.

Homework due Friday at midnight.
Programming assignments due on Sunday nights, with extra office hours that week.
Clicker quizzes for attendance can happen any lecture
Homework!
Combined with the first lab next week; due 1/24.
multilab
nethack
source control
CS 216 has three basic foci:
C++ programming in more depth (and object oriented design, too)
Software engineering techniques
Unix/Linux environment
Why?
Why?
The first skill of an engineer is asking questions.
Active participants
So, programming.
Programming languages are notations.
We express ideas with them
\[ f(x) = x^2 \]
(defun f (x) (* x x))
Declare:
    F(Domain) : Range

Satisfying:
    ! x [Domain] : F(x) = (x * x).
double f(double x)
{
    return x * x;
}
template<typename T>
T f(T x)
{
    return x * x;
}
What ends up being critical is the clarity of expression of those ideas!
$48 \div 2(9+3) =$ ?
Stop wasting everyone’s time and use more parentheses.
“Buffalo buffalo, Buffalo buffalo buffalo buffalo Buffalo buffalo.”
What is engineering?
Definition of ENGINEERING

1 : the activities or function of an engineer

2 a : the application of science and mathematics by which the properties of matter and the sources of energy in nature are made useful to people

b : the design and manufacture of complex products

<software engineering>

3 : calculated manipulation or direction (as of behavior) <social engineering> — compare GENETIC ENGINEERING

See engineering defined for English-language learners »
See engineering defined for kids »

Examples of ENGINEERING

Examples of ENGINEERING

Rhymes with ENGINEERING

fictioneering, mountaineering, power steering
Upper Deviation

Lower Deviation

Fundamental Deviation

International Tolerance Grade

Lower Deviation

Upper Deviation

International Tolerance Grade

Fundamental Deviation

Max. Size

Min. Size

Basic Size

Min. Size

Max. Size
### 4-Band Color Code
- **1st Digit**: 0-9
- **2nd Digit**: 0-9
- **3rd Digit**: 0-9
- **Multiplier**: 1, 10, 100, 1k, 10k, 100k, 1M, 10M
- **Tolerance**: ±5%, ±2%, ±1%, ±0.5%, ±0.25%, ±0.1%
- **Example**: 25kΩ ±5%

### 5-Band Color Code
- **1st Digit**: 0-9
- **2nd Digit**: 0-9
- **3rd Digit**: 0-9
- **Multiplier**: 1, 10, 100, 1k, 10k, 100k, 1M, 10M
- **Tolerance**: ±1%
- **Example**: 460kΩ ±1%

### 6-Band Color Code
- **1st Digit**: 0-9
- **2nd Digit**: 0-9
- **3rd Digit**: 0-9
- **Multiplier**: 1, 10, 100, 1k, 10k, 100k, 1M, 10M
- **Tolerance**: ±10%, ±5%, ±2%, ±1%
- **Example**: 276Ω ±5%

### Temperature Coefficient
- **100ppm**
- **50ppm**
- **15ppm**
- **25ppm**

### Color Code Chart
- **0**: Black
- **1**: Brown
- **2**: Red
- **3**: Orange
- **4**: Yellow
- **5**: Green
- **6**: Blue
- **7**: Violet
- **8**: Gray
- **9**: White

**Example**: The color code for a resistor is **Red-Brown-Green**. This corresponds to 20kΩ with a ±1% tolerance.
I don’t really like to say engineering is largely about failure…
But engineering is largely about failure.
Failure is a common consequence of the interaction between constraint and imperfection.
This is magnified by scale!
Edward Aloysius Murphy, Jr
“Anything that can go wrong, will.”
“If there is any way to do it wrong, he will.”

“If it can be done wrong, then somebody is going to do it wrong.”
Let's apply this to software
Concurrency & Networking
Our most important imperfection is ourselves.
We want software that contains no mistakes.
However – we understand that not making mistakes is not a viable option.
“Writing it perfect the first time” is just not feasible from a cost perspective.
So what do we do?
Anticipate mistakes
Detect mistakes
Fix mistakes
This is software engineering
Incorrect

Correct

“Bad”

Poorly engineered

Well-engineered

“Good”
Why is this so important if it works?
Code is like diamond.
CHAOS Report
Successful: Finished on time and on budget.

Challenged: Completed, but over budget, late, etc.
<table>
<thead>
<tr>
<th>Year</th>
<th>Successful (%)</th>
<th>Challenged (%)</th>
<th>Failed (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>16</td>
<td>53</td>
<td>31</td>
</tr>
<tr>
<td>1996</td>
<td>27</td>
<td>33</td>
<td>40</td>
</tr>
<tr>
<td>1998</td>
<td>26</td>
<td>46</td>
<td>28</td>
</tr>
<tr>
<td>2000</td>
<td>28</td>
<td>49</td>
<td>23</td>
</tr>
<tr>
<td>2004</td>
<td>29</td>
<td>53</td>
<td>18</td>
</tr>
<tr>
<td>2006</td>
<td>35</td>
<td>46</td>
<td>19</td>
</tr>
<tr>
<td>2009</td>
<td>32</td>
<td>44</td>
<td>24</td>
</tr>
</tbody>
</table>
But what does “success” really mean?
Software projects are never really "done"
There’s a transition from active development to maintenance
And maintenance can be over 90% of the time spent on the project!
So how did we get here?
Zuse Z4 - 1945
1968
Go To Statement
Considered Harmful
I could restructure the program's flow or use one little 'goto' instead.

Eh, screw good practice. How bad can it be?

goto main_sub3;

*compile*

[Image of a T-Rex sitting on a chair with a computer on its head]
UNIX®
Celebrating 40 years uptime

1969
THE C PROGRAMMING LANGUAGE

Brian W. Kernighan - Dennis M. Ritchie

1972
IBM PC - 1981
Introducing the same old idea.

YUGO, $3990.
THE C++ PROGRAMMING LANGUAGE

BJARNE STROUstrup

1982
I think C++ is a great teaching language.
This doesn’t mean it’s perfect.
a = b;

a == b;
C++ — by design — is basically a superset of C.
FORTRAN: 1956
C: 1972

1972-1956 = 16
Seers Pacesetters
featuring the latest fashions...
Overalls, Big Tops and Pants

Great Western Shirt
$9.97

1. Trim fit. Finely tailored to emulate authentic Western styling with tailored lapels. Buttoned cuffs. Smooth, wrinkle-resistant polyester and cotton on such a style. For easy care. All-in-one. 2. Long sleeves with three-quarter roll-up sleeves. Long back. For casual wear. Styles have wide, fully adjustable snap-down collars. Tacks out lapels and suede. Long back. 3. A tee shirt styled in the same shirt as a casual shirt. Styles have wide, fully adjustable snap-down collars. Tacks out lapels and suede. Long back. 4. Men's sizes range from S to XL. Women's sizes range from S to XL. For casual shirt. 5. Western shirt with rolled-up sleeves. Styles have wide, fully adjustable snap-down collars. Tacks out lapels and suede. Long back. 6. Men's sizes range from S to XL. Women's sizes range from S to XL. For casual shirt.
Çatalhöyük – c. 7500 BCE
Commercial generation of electricity
Mid to late 19th century
Our field is new.

Software didn’t really exist until 1945, and software engineering started in the 60s.
Edsger Dijkstra
1930-2002