Today

C operators and their precedence
Midterm Review
Homework 2 Questions
## Operator Preference in C (16 levels)

### Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Associativity</th>
</tr>
</thead>
<tbody>
<tr>
<td>( ) [ ]</td>
<td>left to right 16</td>
</tr>
<tr>
<td>. (postfix versions of ++ --)</td>
<td></td>
</tr>
<tr>
<td>(prefix versions of ++ --) sizeof</td>
<td>right to left 15</td>
</tr>
<tr>
<td>! ~ (unary versions of + - &amp; *)</td>
<td>right to left 15</td>
</tr>
<tr>
<td>(type)</td>
<td>right to left 14</td>
</tr>
<tr>
<td>* / %</td>
<td>left to right 13</td>
</tr>
<tr>
<td>+ -</td>
<td>left to right 12</td>
</tr>
<tr>
<td>&lt;&lt; &gt;&gt;</td>
<td>left to right 11</td>
</tr>
<tr>
<td>&lt; &lt;= &gt; &gt;=</td>
<td>left to right 10</td>
</tr>
<tr>
<td>== !=</td>
<td>left to right 9</td>
</tr>
<tr>
<td>&amp;</td>
<td>left to right 8</td>
</tr>
<tr>
<td>^</td>
<td>left to right 7</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>&amp;&amp;</td>
<td>left to right 5</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>?:</td>
<td>right to left 3</td>
</tr>
<tr>
<td>= += -= *= /= %= &amp;= ^= != &lt;&lt;= &gt;&gt;=</td>
<td>right to left 2</td>
</tr>
<tr>
<td>,</td>
<td>left to right 1</td>
</tr>
</tbody>
</table>
### ++ and --

Unary increment(++)/decrement(--) operators

Prefix (to left, before): --x     decrement first, then use

Postfix (to right, after): x++   use first, then increment

```plaintext
x = 3;
y = x++; // y gets 3, then x incremented to 4
z = --x; // x decremented to 3, then z gets 3
// x, y, and z all are 3 at end
```

```c
int j;
int ni = n*i;
double *rowp = a+ni;
for (j = 0; j < n; j++)
    {*rowp = b[j]; rowp++;}
```

```c
int j;
int ni = n*i;
double *rowp = a+ni;
for (j = 0; j < n; j++)
    *rowp++ = b[j];
```
Precedence Examples

\[ a \times b + c \]
\[ a - b + c \]
\[ \text{sizeof(int)} \times p \]
\[ *p \rightarrow q \]
\[ *x++ \]
\[ a = b ++ \]
\[ a ++ b \]
\[ a +++ b \]
\[ a ++++ b \]
\[ a +++++ b \]
Precedence Examples

- \( a \times b + c \) vs. \( (a \times b) + c \)
- \( a - b + c \) vs. \( (a - b) + c \)
- \( \text{sizeof}(\text{int}) \times p \) vs. \( (\text{sizeof}(\text{int})) \times p \)
- \( p \rightarrow q \) vs. \( *(p \rightarrow q) \)
- \( x \rightarrow ++ \) vs. \( *(x++) \) not \( *(x)++ \) incr after use
- \( a + = b ++ \) vs. \( a + = (b++) \) incr after use
- \( a + ++ b \) vs. \( a + (+b) \)
- \( a +++ + b \) vs. \( (a++) + b \) not \( a + (++b) \) incr after use
- \( a +++++ + b \) vs. \( (a++) + (+b) \) incr after use
Midterm Review - Topics

- Memory + Data
- Numerical Representations
  - Integers, Floats, Doubles
- Introductory C
  - Operators, Pointers
  - C => Machine Code
- x86 Assembly
  - Basics, Conditionals, Iteratives
- Stack & Procedures
- Data Structures
  - Arrays (of var. dimension), structs, unions
- Buffer Overflows
Midterm Review - Assignments

- HW 0: Intro Performance
- Lab 1: Bitwise Operations / Basic C
- HW 1: Numerical Representations + x86 ISA
- Lab 2: x86 + Debugging
- HW 2: More x86

- HWs cover concepts
- Labs reinforce concepts + provide experience
- Content from both fair game!