OOP – The conclusion

Lecture 25

Quiz

• 50%+ on HW4 material.
• 25% on Matlab (basics done in class and lab)
• <25% on simple OOP material from last few days.
• Maybe basic code reading question
  – Like quiz1

HW4 notes

• As per e-mail (and class discussion) recall that like addition and multiplication AND (*) binds tighter than OR (+)
• When adding two 8-bit fixed-size numbers, the result must be the same size
  – Overflow is when the result is out of the range of representation.
• HW4 answers posted.
  – Typo in answer to last truth table on page 3. Will post fix by noon today.

Where we are

• From a programming language viewpoint, there are only a few things left to introduce
  – Vectors
  – Enumerated types
  – Pointers (only a little)
  – Operator overloading
• And a few things to spend time to clarify/expand on
  – Organization of a program
    • (Using header files, function prototypes)
  – Characters
### Non-C++ language things

- Data structure design
  - Use of stacks and queues
  - Smart arrays
- Some more on complexity
  - Better sorts
- Matlab
  - Much more on using it.

### Misc.

- E-mail
  - I'm behind.
  - I should be able to catch up today

### Today

- Another shot at complex numbers and classes
- Why classes
- Some stuff on characters
  - Bits is bits

### Code example
Why classes? (again)

- Consider our Cmpx code
  - The class is self-contained (like our string class)
- But not the best of all possible examples, because data makes sense to be able to access directly.
  - Let’s work on designing a “time” class.
  - Want time to be stored in hours, minutes and seconds
  - Want to be able to add and subtract time.
  - What to be able to ask user for time and print time.
  - At all times want to be sure sec<60 and min<60

Chars

- ASCII
  - Simply a mapping of 8-bit chars to certain symbols.

<table>
<thead>
<tr>
<th>Base10</th>
<th>Base 2</th>
<th>symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>097</td>
<td>01100001</td>
<td>a</td>
</tr>
<tr>
<td>098</td>
<td>01100010</td>
<td>b</td>
</tr>
<tr>
<td>099</td>
<td>01100011</td>
<td>c</td>
</tr>
<tr>
<td>100</td>
<td>01100100</td>
<td>d</td>
</tr>
<tr>
<td>101</td>
<td>01100101</td>
<td>e</td>
</tr>
<tr>
<td>102</td>
<td>01100110</td>
<td>f</td>
</tr>
</tbody>
</table>

Note

- int bob=‘a’;
  - bob is 97.
- int bob=‘a’+3
  - bob 100.
- char bob=‘a’+3;
  - bob is still 100 but..
    - If you print it you get a ‘d’