CMPSC112
Lecture 5: Inheritance

Prof. John Wenskovitch
09/07/2015
Last Time

• Design of three classes: Human, CPU, and Game
• Implementation of the Human class
• Implementation of the CPU and Game classes
## Discovered Issue

- The Human and CPU classes shared a bunch of functions. Lots of extra typing.

<table>
<thead>
<tr>
<th>Human</th>
<th>Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>- userScore: int</td>
<td>- computerScore: int</td>
</tr>
<tr>
<td>+ Human()</td>
<td>+ Computer()</td>
</tr>
<tr>
<td>+ getScore(): int</td>
<td>+ getScore(): int</td>
</tr>
<tr>
<td>+ getInput(): String</td>
<td>+ getInput(): String</td>
</tr>
<tr>
<td>+ incrementScore()</td>
<td>+ incrementScore()</td>
</tr>
</tbody>
</table>
Inheritance

• **Definition:** A programming technique or mechanism for creating a hierarchy of classes.

• **Practical approach:** Put everything that is shared in a *parent class*, and leave everything that is different in the *child classes*. Give the original class access to things in the new parent class.
What Does the Subclass Do?

• The subclass automatically inherits all methods from the superclass.
• The subclass can **augment** the superclass by adding new variables and new methods.
• The subclass can also specialize superclass behaviors by providing a new implementation that **overrides** an existing method.
Inheritance

```
Player
- score: int
+ getScore(): int
+ incrementScore()
```

```
Human
+ Human()
+ getInput()
```

```
CPU
+ Computer()
+ getInput()
```

“Parent class”
“Base class”
“Super-class”

“Child class”
“Sub-class”
Accessing Variables

- **Accessor / “getter”** – Public function that provides read access to a private variable.
- **Mutator / “setter”** – Public function that provides write access to a private variable.
Any Questions?