Decisions
Lecture 4 (7/6/2006)

Useless Fact of the Day
• The last FIFA World Cup was watched by 1.1 billion people (1/6 the world population), making it the most widely-viewed sporting event in the world (surpassing even the Olympic Games).

Topics
• The if statement
• AND and OR
• Comparison and Equality

The if Statement
if (A)
{
we do this stuff if statement A was true.
}
else if (B)
{
we do this stuff if A was false but B was true.
}
else
{
we do this stuff if both A and B were false.
}
**AND and OR**

- We can combine multiple boolean statements into one
- **AND:** `&&`
  - The statement `A && B` is true if and only if the individual statements `A` and `B` are both true
- **OR:** `||`
  - The statement `A || B` is true if and only if at least one of the individual statements `A` and `B` is true
- Example:
  - `5 == 7 && 7 == 7` → false
  - `5 == 7 || 7 == 7` → true
  - `(5 == 7 && 7 == 7) || 4 == 4` → true

**Comparing doubles**

- Why shouldn’t you use `==` with doubles?
  - Round-off error makes things imprecise
  - `12.999999999999998` does not equal `13`
- Instead of `==`, we must compare within a tolerance
  - We assume doubles, `x` and `y`, to be equal if they are within some small tolerance of each other
- Instead of using `x == y`, use something like one of these two things:
  - `x - y < 0.0000005 && x - y > -0.0000005`
  - `Math.abs(x - y) < 0.0000005`

**Comparing Strings**

- Why can’t you use `==` with Strings (or other objects)?
  - `==` checks to see if two things are the same object
  - `==` does not check the contents of the objects to see if they are the same
- Two Strings, then, can store the same data (“hello”, for example), but since they are different objects, `==` returns false when comparing them
- Use the `String.equals(String)` method (an instance method)
- Instead of asking `string1 == string2`, use:
  - `string1.equals(string2)`

**Equality**

- When using the `===` operator is awesome:
  - comparing integers (Ex: `13 == 13` → true)
  - comparing characters (chars) (Ex: `'a' == 'b'` → false)
  - comparing booleans (Ex: `ball.isVisible() == star.isVisible()`)
- When using the `===` operator will can be very bad:
  - comparing doubles
    - Ex.: `(Math.sqrt(13) * Math.sqrt(13)) == 13` → false
  - comparing Strings or other objects
    - Ex.: `String one = "wtf"; String two = "wt";
      two = two + "f";
      two == one` → false
Comparing Strings

• Other useful String methods:
  • startsWith(String prefix)
  • equalsIgnoreCase(String s)
  • compareTo(String s)
    • compares two strings lexicographically (Ex.: “cat” > “cab”)
    • result is a negative number if the string is less than s, positive if it is greater, and zero if they are equal
  • toLowerCase() and toUpperCase()
  • substring(int beginIndex) and substring(int beginIndex, int endIndex)
  • returns a new substring which is a substring of this string

Game-Relevant Decisions

• Number comparison
  • Is the number of bad guys left zero? Is the number of points greater than 1000? Is the current level greater than 3?

• Intersection
  • Is the bullet intersecting the space ship?
    • Often simplified to: Is the point within the rectangle?
  • Are the two space ships intersecting?
    • Often simplified to: Are the two rectangles/circles intersecting?