

Lecture 34: Big Programs/Searching

CS 51G
Spring 2018
Kim Bruce

Announcements

- Nibbles lab Friday

Writing a big Program

- Identify the objects to be modeled in your program.
- For each type of object identified:
 - List its properties.
 - List its behaviors.
- Model properties with instance variables.
- Model behaviors with methods. Init. code too.
 - Focus on the method headers & parameters.
 - What will be the result of each method invocation?

Testing & Debugging

- Once find location of bug, relatively easy to fix.
- Test small (simple) pieces of code.
 - When put together, be confident details work

Writing & Testing Simon

- Pop up window with buttons
- Does pressing one button work (no song)
- Create and play songs with 1 or 2 notes.
 - Add new note after play
- Start onMousePressed

Testing

- After each part, thoroughly test
 - If can't see what happens, add print statements to show what has changed.
 - Often best strategy is to write a test harness which only designed to test program pieces.
 - As I did for NibbleField!!
 - If can't find error, comment out parts until find where error arises

Nibbles Development

- Write missing methods (one at a time) in NibbleField and test with NibbleFieldTester
- Construct snake of length 1
 - Write shrink and stretch methods
 - Makes snake of length 1 move in straight line
 - Dies when hits edge
 - Steer snake using direction:=() method
 - Do same with snake of length 3

Nibbles Development

- Worry about eating!
 - Detect food before running over it.
 - Skip shrinking 3 times for each food item eaten.
 - I.e., if eat 2 in a row will stretch 6 times before shrinking.
 - If you decide to stretch an extra 3 times, you will NOT succeed because you must watch out for walls, etc while stretching (and might eat again)
 - *Follow this advice or waste many hours of your time!!!*
- Worry about dying by biting itself.
 - check out-of-bounds before checking biting itself

Searching

- Looking in a collection of values for some specific value (where is the 17 in this array of int?).
- Looking for a value with a specific property (which object on the canvas contains the location where I clicked the mouse?).
- Looking for a record in a database (what is the tax history for the last four years for the taxpayer with SSN 101-11-1009?).
- Searching for text in some document or collection of documents (what web pages contain the text “searching and sorting algorithms?”).
- What known amino acid sequences best match this sequence gathered from proteins in the SARS virus?

Sample Code

```
// Does the scribble contain pt?  
method contains (pt: Point) -> Boolean {  
  for (scribbleLine) do {nextLine: Line ->  
    if (nextLine.contains (pt)) then {  
      return true  
    }  
  }  
  false  
}
```

List

```
method contains (pt: Point) -> Boolean {  
  first.contains (pt) || rest.contains (pt)  
}
```

Recursive

Searching

- Iterative vs Recursive
- Linear vs Binary.
 - Binary requires list be sorted!
- How many comparisons does it take to find an element?
- <http://www.cs.pomona.edu/classes/cs051G/demos/SearchSort/search.grace>

Questions?