Designing Test Cases

CSC148, Introduction to Computer Science
Fall 2016
Diane Horton
Example

• A function to find the maximum in a list.
• These test cases:

<table>
<thead>
<tr>
<th>List</th>
<th>Expected Result</th>
<th>Test passed?</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3, 6, 4, 42, 9]</td>
<td>42</td>
<td>yes</td>
</tr>
<tr>
<td>[22, 32, 59, 17, 18, 1]</td>
<td>59</td>
<td>yes</td>
</tr>
<tr>
<td>[1, 88, 17, 59, 33, 22]</td>
<td>88</td>
<td>yes</td>
</tr>
<tr>
<td>[1, 3, 5, 7, 9, 1, 3, 5, 7]</td>
<td>9</td>
<td>yes</td>
</tr>
<tr>
<td>[7, 5, 3, 1, 9, 7, 5, 3, 1]</td>
<td>9</td>
<td>yes</td>
</tr>
<tr>
<td>[561, 1024, 13, 79, 97, 4]</td>
<td>1024</td>
<td>yes</td>
</tr>
<tr>
<td>[9, 6, 7, 11, 5]</td>
<td>11</td>
<td>yes</td>
</tr>
</tbody>
</table>
Example

• Are you confident the function works?
• What if I said it passed 20 more test cases?
• What if I said it passed 100 more test cases?

• I hope you are skeptical.
• Quantity of test cases means little.
• *Quality* matters.
(In reality there are many more possible test cases)
Are we convinced?

- We want to believe the function works on *every possible input*.
- But the test cases seem to be chosen at random.
- We have no reason to believe the function works in the untested cases.
- And we can’t test them all!
A better approach

• Carve the possible inputs into meaningful categories.

• Pick a representative from each category to test.

• If we choose the category well
  – it will be reasonable to extrapolate from one test case to all other test cases in the category.
(In reality there are many more possible test cases)
tested

untested but reasonable to extrapolate
How to define the categories

• We base the categories on relevant properties of the inputs.

• Examples:
  – Size of something: 0, 1, larger
  – Size: even, odd
  – Position of something: beginning, ending, elsewhere
  – Relative location of two things: adjacent, separated
  – Presence of duplicates: yes, no
  – Ordering: unsorted, non-decreasing, non-increasing
  – Values of an integer: all negative, all positive, mixed
  – Etc.!!!
But which properties?

• We have to decide on which properties are relevant – there are so many.
• We decide based on knowing what the function or method does.
• If we know how it does it, that can influence our choices also.
  – For example, if a method divides a list in half, odd vs even size is pretty important!