CS 1110
Prelim 1 Practice/Review Session

http://www.cs.cornell.edu/courses/cs1110/2021sp
Announcements

- A3 due Sun Mar 28
- Prelim 1 Tues Mar 30 at 6:30pm in-person (university-scheduled)
  - Check CMS for your exam info if you requested alternate time/format
  - In-person: Bring pens/pencils/erasers (bring several). Bring a watch or even an actual clock if you have one. No smart watches/phones! You may not be able to see the wall clock in Barton from your seat. **Bring Cornell ID.**
  - Online: Your proctor will contact you about a mock exam. You **must** do the mock exam to be allowed to write the actual exam.
- Read Prelim 1 Study Guide. *Note spring different from fall.*
- Tues Mar 30 lecture and lab time → office hours
- Wedn Mar 31 no labs (so no new lab exercises next week)
Exam Topics

• String slicing functions
• Call frames and the call stack
• Functions on mutable objects
• Testing and debugging
• Conditionals
• Lists and simple iteration

Today:
• Start with lists and iteration—not in posted old review slides
• Testing and debugging
• Other topics if time allows

Dictionaries not on Prelim 1
def count_non_space_chars(myList):
    """Returns: number of non-space characters in the strings in myList.
    Example: count_non_space_chars(['U', 'r', '', 'gr8']) returns 5
    Precondition: myList is a list of strings. Each string in myList can
    contain only spaces, letters, digits."""
def inflate(myList, p_percent):
    """Inflate each number in myList by p_percent while maintaining the type (int or float). For any int in myList, round down the inflation. Precondition: myList is a list of positive numbers (int and/or float). Precondition: p_percent is a positive number (int or float)."""

An example:

```python
>>> aList= [100, 100.0, 1, 1.0]
>>> p= 1.6
>>> inflate(aList,p)
>>> aList
[101, 101.6, 1, 1.016]
```
def inflate(myList, p_percent):
    """Inflate each number in myList by p_percent while maintaining the type (int or float). For any int in myList, round down the inflation. 
Precondition: myList is a list of positive numbers (int and/or float). 
Precondition: p_percent is a positive number (int or float)."""
def before_space(s):
    """Returns: the substring before the first space character in string s.
    Precondition: string s contains at least one space."""

Come up with at least three distinct test cases. Write the test input, expected output, and rationale.
What should I be testing?

**Common Cases:** typical usage

**Edge Cases:** live at the boundaries

- **Target location in list:** first, middle, last elements
- **Input size:** 0, 1, 2, many (length of lists, strings, etc.)
- **Input Orders:** e.g., max(big, small), max(small, big)…
- **Element values:** negative/positive, zero, odd/even
- **Element types:** int, float, str, *etc.*
- **Expected results:** negative, 0, 1, 2, many

*Not all categories/cases apply to all functions.*

*Use your judgement!*
Functions on Objects

- **Class**: `Rect`
  - **Constructor function**: `Rect(x, y, width, height)`
  - Remember constructor is just a function that gives us an object of that type and returns its identifier

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>x</code></td>
<td>float, x coord of lower left corner</td>
</tr>
<tr>
<td><code>y</code></td>
<td>float, y coord of lower left corner</td>
</tr>
<tr>
<td><code>width</code></td>
<td>float, $&gt;0$, width of rectangle</td>
</tr>
<tr>
<td><code>height</code></td>
<td>float, $&gt;0$, height of rectangle</td>
</tr>
</tbody>
</table>
def move(r, xc, yc):
    """Set the attributes of Rect `r` such that its center lies on the x- and y-coordinates `xc` and `yc`, respectively.
    Precondition:  `r` is a Rect object.
    Precondition:  `xc`, `yc` are each a float."""
Good Luck!