

More looping

Announcements

- Nice job on the quiz!
- Lab 3 posted
- After this week we'll have most of the basics

Today's plan

- Hand back quizzes
- Review practice problems from Friday
- `while` loops: repeat indented lines of code until a condition becomes `False`

Testing parity

- Use the **remainder** operator, %, to test a number's parity, i.e. whether it is even or odd.
- If a number has remainder 0 when divided by 2 it is even.
- If a number has remainder 1 when divided by 2 it is odd.

Testing parity

```
n = int(raw_input("Enter number: "))

if n % 2 == 0:
    print("%d is even" % n)
else:
    # n % 2 is either 0 or 1 so
    # if we reach here, n % 2 == 1
    print("%d is odd" % n)
```

Truth Tables

- We can use truth tables to verify rules of logic like De Morgan's laws:

A	B	not (A or B)	(not A) and (not B)
T	T	F	F
T	F	F	F
F	T	F	F
F	F	T	T

More on `if-elif-else`

- Within a single conditional statement (`if`, possibly followed by one or more `elif`'s and possibly followed by one `else`) at most one block of indented code will execute.
- Each appearance of `if` starts a new conditional statement.

Compatibility Quiz

- The program uses:
 - Indexing within a for loop
 - String formatting
 - Accumulator
 - Parallel lists of strings
 - Conditionals

while loops

- Syntax:

```
while <boolean expression>:
```

```
    <block of code>
```

- Semantics: repeat the block of code as long as the <boolean expression> evaluates to True. If the block of code does nothing to change the boolean expression, the while loop will repeat indefinitely.

See you Wednesday!