Objectives

- Review `while` loops

- Lab 3 Feedback, Discussion

- Lab 4
  - ➢ Two fewer problems (80 points)

Nondeterministic Decisions

- Sometimes, we don't want to necessarily know that a specific decision is always made

- For example, games often use randomness to make decisions

  - Roll dice
  - Coin flips
  - Location and behavior of baddies

Flipping Coins

- Simulate by randomly selecting between 0 (heads) and 1 (tails)

- Program: `coinFlip.py`

- Problem: How many flips does it take to get 3 consecutive heads?

consecutiveHeads.py

Lab 3 Feedback

- More difficult problems
- Took off for insufficient testing
- Took off for poor naming, style, comments

Beatles Lyrics Problem

- Using two `for` loops, a variable with value "She loves you," and another variable with value "yeah", print out the Beatles lyrics…

```python
sly = "She Loves You,"
yeah = "yeah"

#prints the line three times
for line in xrange(3):
    print sly,

    # print 2 yeahs, with commas
    for ycount in xrange(2):
        print yeah + ",",

    print yeah # closes the fence post

print "Yea-aahh!"
```

Common Issue

- Using loops for statements that should only be executed once

```python
#prints the line three times
for line in xrange(3):
    sly = "She Loves You,"
```

What is the difference between executing this in the loop and just executing once outside the loop?

Common Issue

- Using loops for statements that should only be executed once

```python
#prints the line three times
for line in xrange(3):
    sly = "She Loves You,"
```

What is the difference between executing this in the loop and just executing once outside the loop?

Result is the same
But in loop, executes two more times, not changing
    extra, unnecessary assignments

Common Issue

- Using same variable for different purposes
  - ➤ Will cause problems/confusion later

```
game = 0
for x in xrange(NUMSIMS+1):
    x = random.randint(MIN, MAX)
    game += 1
```

Beyond that we don't need the *game* variable…

The purpose of x has changed:
    x is the counter for the loop
    x is the random number for determining who won

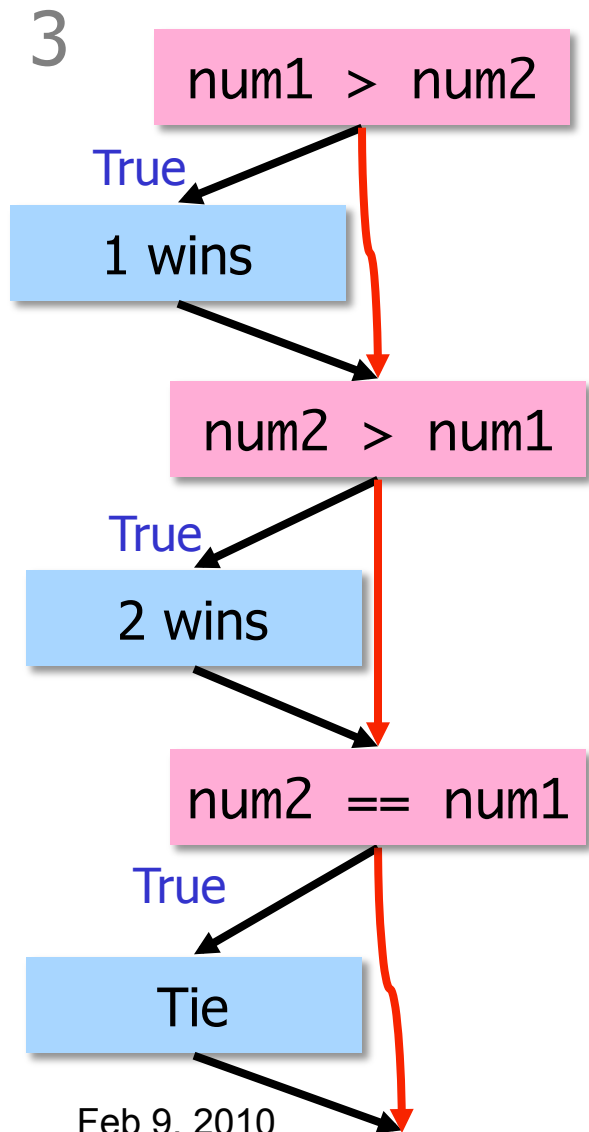Separate purposes → separate variables

Common Issue

- Instead of

```
game = 0
for x in xrange(NUMSIMS+1):
    x = random.randint(MIN, MAX)
    game += 1
```
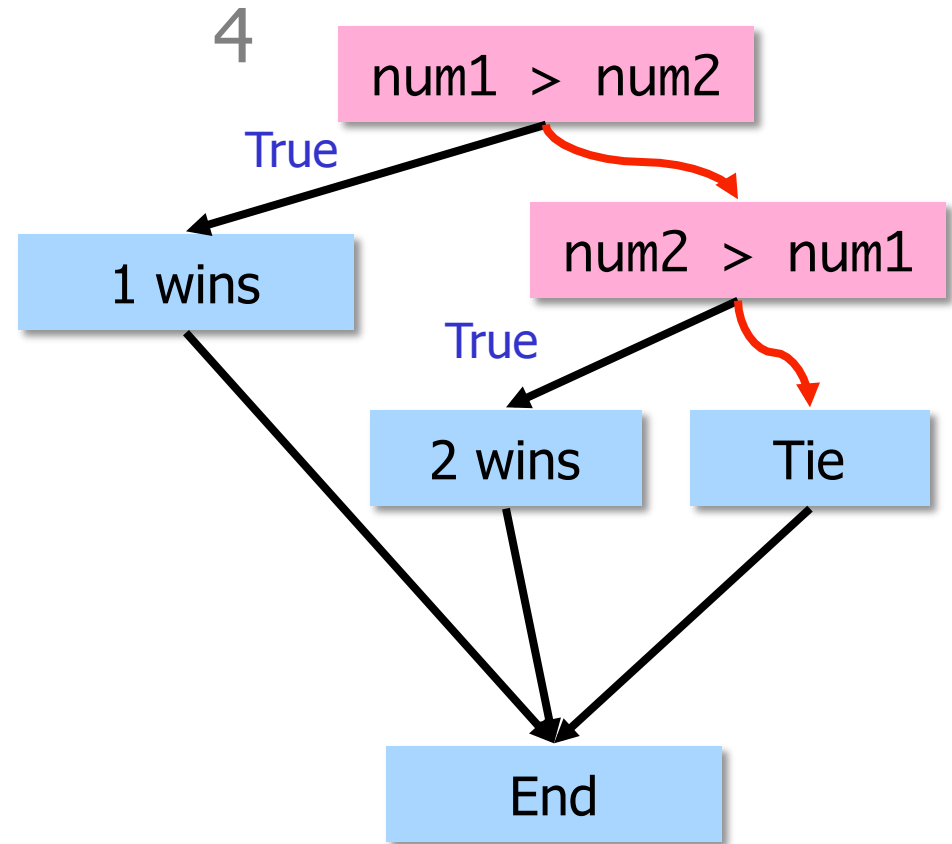
- This is better:

```
for game in xrange(NUMSIMS+1):
    outcome = random.randint(MIN, MAX)
```
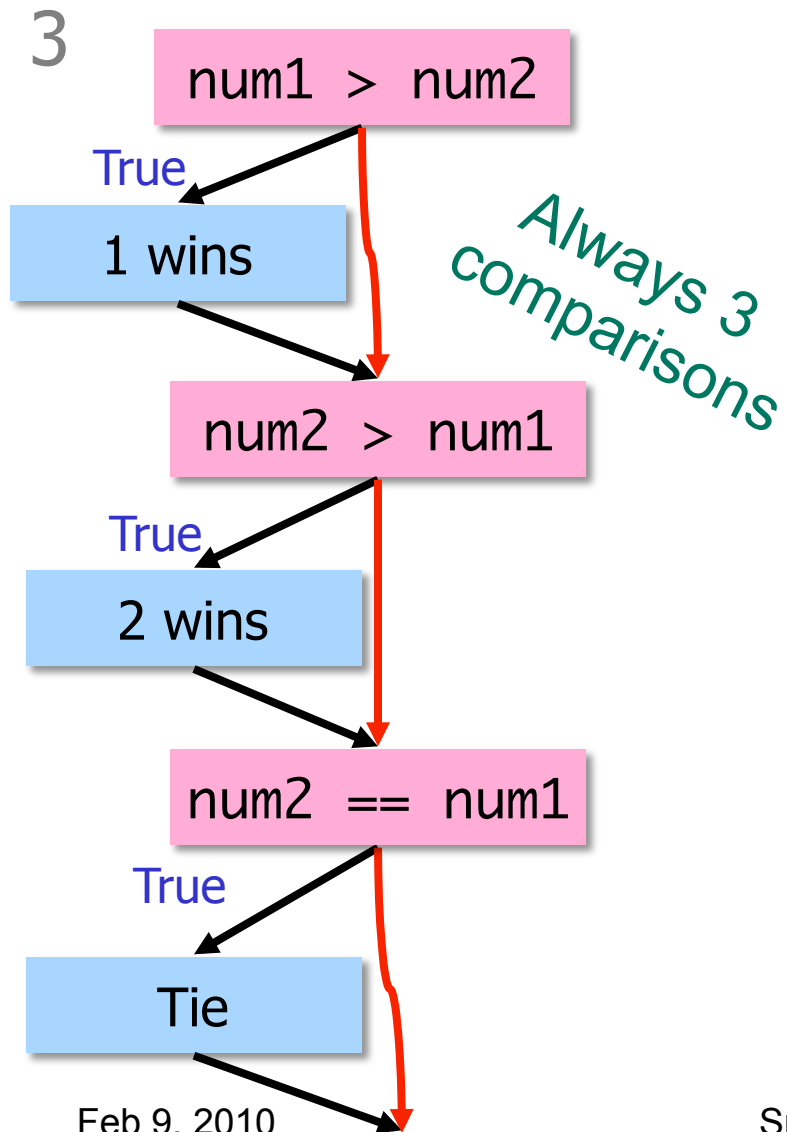
Problem 3, 4 Efficiency
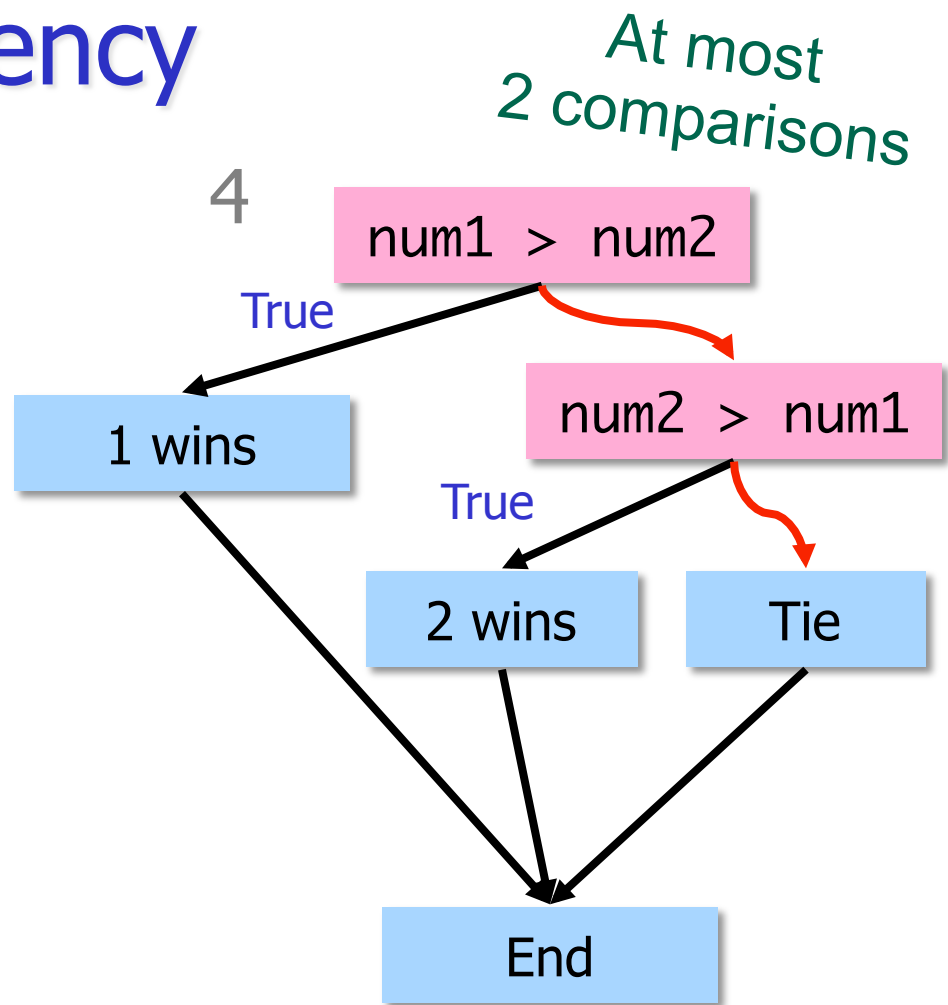
3

num1 > num2

True

1 wins

num2 > num1

True

2 wins

num2 == num1

True

Tie

4

num1 > num2

True

1 wins

num2 > num1

True

2 wins

Tie

End

- How many conditions evaluated?
- Consider case num1=5, num2=3

Problem 3, 4 Efficiency

At most
2 comparisons

3

num1 > num2

True

1 wins

Always 3
comparisons

num2 > num1

True

2 wins

num2 == num1

True

Tie

4

num1 > num2

True

1 wins

num2 > num1

True

2 wins

Tie

End

Sprenkle - CSCI111

Problem 4 Efficiency

Which tends to be more efficient?
How many conditions to evaluate?

num1 == num2

True

Tie

num2 > num1

True

2 wins

1 wins

End

num1 > num2

True

1 wins

num2 > num1

True

2 wins

Tie

End
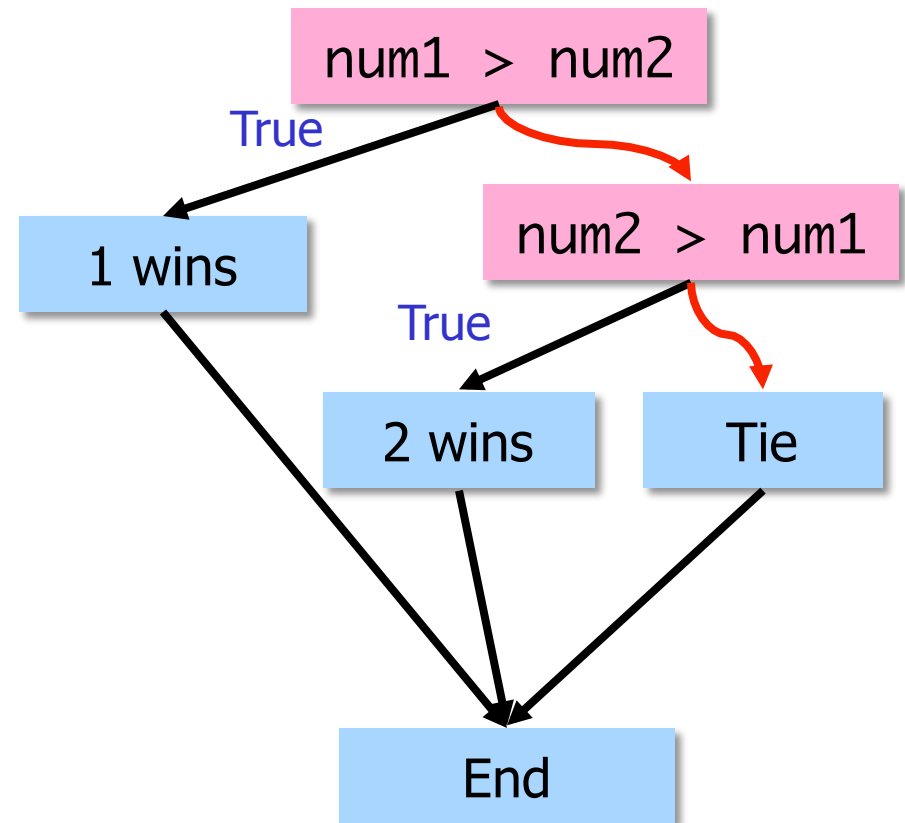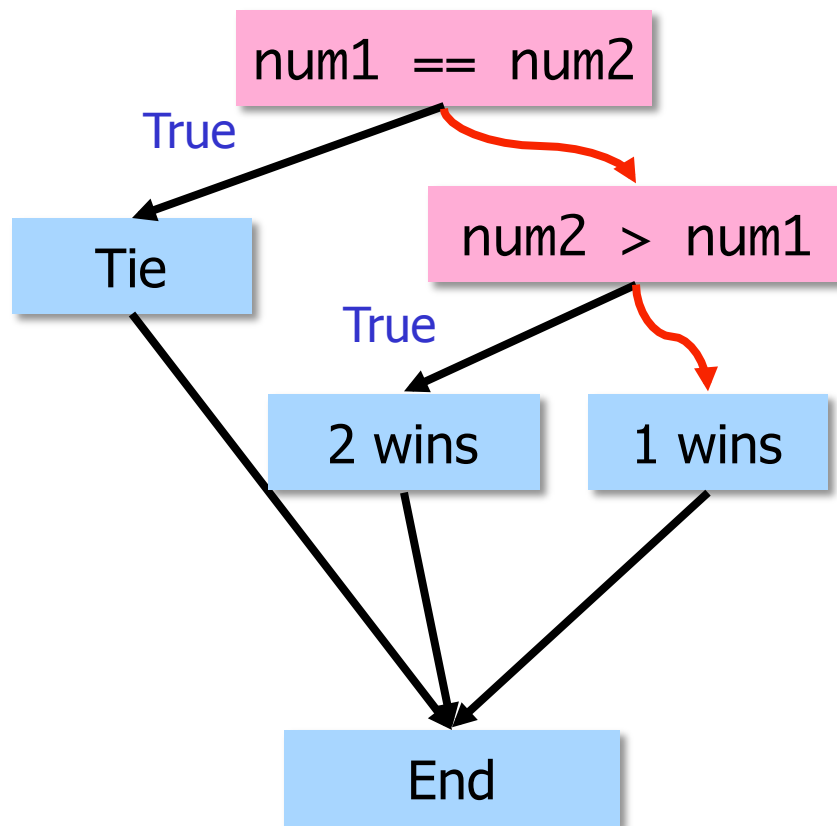
Lab 4 Overview

- Practice Python programming
  - ➢ Advanced `for` loop
  - ➢ Use `sys` module
  - ➢ Indefinite loops (`while`)
- Due on Friday before class