Introduction to Computer Programming

Lecture 8
REVIEW
Logical Operators (*and*, *or*, *not*) combine Boolean expressions

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A and B</strong></td>
<td>True if both A and B are true, otherwise false</td>
</tr>
<tr>
<td><strong>A or B</strong></td>
<td>True if either A or B are true, otherwise false</td>
</tr>
<tr>
<td><strong>not A</strong></td>
<td>True if A is false, false if A is true</td>
</tr>
</tbody>
</table>
Practice: Boolean Logic

5 > 7 or 8 <= 10

5 >= 5 and 6 > 6

X or (not X)

X and (not X)
Practice: Examples

5 > 7 or 8 <= 10  # True

5 >= 5  and  6 > 6

X  or  (not X)

X  and (not X)
Practice: Examples

5 > 7 or 8 <= 10  # True

5 >= 5 and 6 > 6  # False

X or (not X)

X and (not X)
Practice: Examples

5 > 7 or 8 <= 10  # True

5 >= 5 and 6 > 6  # False

X or (not X)  # Always true

X and (not X)
Practice: Examples

5 > 7 or 8 <= 10 # True

5 >= 5 and 6 > 6 # False

X or (not X) # Always true

X and (not X) # Always false
Another Example – True or False?

10 > 5 and 12 < 13 or 3 > -3 and 5 != abs(-5)
Example – True or False?

10 > 5 and 12 < 13 or 3 > -3 and 5 != abs(-5)

T and T or T and F
Another Example – True or False?

10 > 5 and 12 < 13 or 3 > -3 and 5 != abs(-5)
  T and T or T and F
  T or F

“and” has higher precedence than “or”
Therefore, evaluate “and” first
Another Example – True or False?

10 > 5 and 12 < 13 or 3 > -3 and 5 != abs(-5)

T and T or T and F

T or F

True
Random Numbers

(beginning of Chapter 6 in Gaddis)

“Pick a number, any number, …”
Why Do We Need Random Numbers?

- Randomize your play list
- Randomize your screen-saver
- Randomize alien attacks in your favorite game
- Shuffle the cards for solitaire
- Simulating the real world
- .....
Generating Random Numbers

• Easy for humans
  – Toss a coin
  – Roll dice
  – Pick a raffle number from a hat

• Very hard for Computers
  – Computers can only be pseudo-random
  – Hard mathematical problem
What is a good random number generator?
Properties of a good random number generator

• It must be unpredictable (i.e. random)
• It must be fair

Taking turns is fair, but predictable

Loaded dice are unpredictable, but not fair
Random Numbers in Python

# Generate a random number in the range of 1 – 100

import random

...  
my_num = random.randint(1, 100)  
print(my_num)
Decision Trees for modeling If statements

Question1?

- yes
  - yes
    - Answer1
  - no
    - no
    - Answer2
    - Answer3
Loan Qualification
make more than $50,000

Salary $50,000?

yes
Get a Loan

no
No Loan
.... And same job for at least 3 years

- $\text{Salary} \geq $50,000?
  - yes
  - no
  - $\text{Years worked} \geq 3$?
    - yes
    - no
    - Get Loan
    - No Loan
  - No Loan
Or have a salary $\geq 100,000$

- $Salary \geq 50,000$?
  - yes
    - $Years \text{ worked} \geq 3$?
      - yes
        - Get Loan
      - no
        - No Loan
  - no
    - No Loan

- $Salary \geq 100,000$?
  - yes
    - Get Loan
  - no
    - No Loan
Example: Bank Marketing

Salary >= $50,000?

- yes
- no

Salary >= 100,000 ?

- yes
- no

- Offer loan at good interest rate
- Offer special promotion

Have a nice day!