CSc 110, Spring 2017

Lecture 12: Random Numbers

Adapted from slides by Marty Stepp and Stuart Reges

http://xkcd.com/221/
Randomness

• Lack of predictability: don't know what's coming next

• Random process: outcomes do not follow a deterministic pattern (math, statistics, probability)

• Lack of bias or correlation (statistics)

• Relevant in lots of fields
  • Genetic mutations (biology)
  • Quantum processes (physics)
  • Random walk hypothesis (finance)
  • Cryptography (computer science)
  • Game theory (mathematics)
  • Determinism (philosophy)
Pseudo-Randomness

• Computers generate numbers in a predictable way using a mathematical formulas

• Parameters may include current time, mouse position
  • In practice, hard to predict or replicate

• True randomness uses natural processes
  • Atmospheric noise (http://www.random.org/)
  • Lava lamps (patent #5732138)
  • Radioactive decay
The Random class

- random functions generate pseudo-random numbers.
  - Class random is found in random
    from random import *

<table>
<thead>
<tr>
<th>function name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>random()</td>
<td>returns a random float in the range ([0, 1]) in other words, 0 inclusive to (max) exclusive</td>
</tr>
<tr>
<td>randint((min, max))</td>
<td>returns a random integer in the range ([min, max]) in other words, (min) to (max) inclusive</td>
</tr>
</tbody>
</table>

- Example:

```python
from random import *
random_number = randint(1, 10)  # 1-10
```
Generating random numbers

• To get a number in arbitrary range $[min, max]$ inclusive:

  $\text{randint}(min, max)$

  • Where size of range is $(max - min)$

• Example: A random integer between 4 and 10 inclusive:

  $n = \text{randint}(4, 10)$
Random and other types

- `random` function returns a float between 0.0 - 1.0
  - Example: Get a random GPA value between 1.5 and 4.0:
    ```python
    random_gpa = random() * 2.5 + 1.5
    ```

- `randint(a, b)` function returns an integer in the given range
  - Example code to randomly play Rock-Paper-Scissors:
    ```python
    r = randint(0, 2)
    if (r == 0):
        print("Rock")
    elif (r == 1):
        print("Paper")
    else:  # r == 2
        print("Scissors")
    ```
Random question

• Write a program that simulates rolling two 6-sided dice until their combined result comes up as 7.

  2 + 4 = 6
  3 + 5 = 8
  5 + 6 = 11
  1 + 1 = 2
  4 + 3 = 7
  You won after 5 tries!
# Rolls two dice until a sum of 7 is reached.
From random import *

def main():
    tries = 0
    sum = 0
    while (sum != 7):
        # roll the dice once
        roll1 = randint(1, 6)
        roll2 = randint(1, 6)
        sum = roll1 + roll2
        print(str(roll1) + " + " + str(roll2) + " = " + str(sum))
        tries = tries + 1
    print("You won after " + str(tries) + " tries!")
Random question

• Write a program that plays an adding game.
  • Ask user to solve random adding problems with 2-5 numbers.
  • The numbers to add are between 1 and 10
  • The user gets 1 point for a correct answer, 0 for incorrect.
  • The program stops after 3 incorrect answers.

4 + 10 + 3 + 10 = 27
9 + 2 = 11
8 + 6 + 7 + 9 = 25
Wrong! The answer was 30
5 + 9 = 13
Wrong! The answer was 14
4 + 9 + 9 = 22
3 + 1 + 7 + 2 = 13
4 + 2 + 10 + 9 + 7 = 42
Wrong! The answer was 32
You earned 4 total points
Pseudo-code

• Main program

  while the player has lost < 3 games
    play a game  ( must get a result back)
    if player lost
      add to losers
    else
      add to winners
  print the total points earned
Pseudocode to code...
# Asks the user to do adding problems and scores them.
from random import *

def main():
    # play until user gets 3 wrong
    points = 0
    wrong = 0
    while (wrong < 3):
        result = play()  # play one game
        if (result == 0):
            wrong += 1
        else:
            points += 1

    print("You earned " + str(points) + " total points.")
Pseudo-code

• Play a game

  get the random number of operands from 2 to 5
  initialize the sum
  print the sum (lay the post !)

  for the number of operands
    get a random number from 1 to 10
    add it to the sum
    print "+" and the random number
  print "="

  prompt for the user's guess
  if guess is correct
    return 1
  else
    print out message to user with correct answer
    return 0

4 + 10 + 3 + 10 = 27
9 + 2 = 11
8 + 6 + 7 + 9 = 25
Wrong! The answer was 30
Pseudocode to code...
def play():
    operands = randint(2, 5)
    sum = randint(1, 10)
    print(sum, end='')
    for i in range(2, operands + 1):
        n = randint(1, 10)
        sum += n
        print(" + " + str(n), end='')
    print(" = ", end='')

    guess = input()
    if (guess == sum):
        return 1
    else:
        print("Wrong! The answer was " + str(total))
        return 0