COMP 141

Loops that count

Announcements

• Reminders
  – Program 4 due Sunday, October 1st by 11:55pm

Pseudocode

• an informal way of writing algorithms for humans to read (not computers!)
• Illustrates the logic of an algorithm, but omits details that people can fill in automatically.
• You get to make it up as you go along, as long as you (and other people) can easily understand it.

• Instead of saying
  name = input("What is your name? ")

• Pseudocode might use a line that says
  name = ask user for name
• Instead of saying
  
  \[
  \text{if } x \geqslant 0 \text{ and } x \leqslant 100: \\
  \quad \text{print(“$%.2f” } \% \, x) \\
  \]

• Pseudocode might use a line that says
  
  \[
  \text{if } x \text{ is between } 0 \text{ and } 100:\n  \quad \text{print } x \text{ with 2 decimal places} \\
  \]

The point is to get your ideas down on paper quickly, so you can worry about the programming details and exact syntax later.

To write any while loop:
1. Write out pseudocode for what the loop does, explicitly repeating lines until you've repeated the code at least twice.
2. Include an "if" statement in your code that will be True if you want the loop to keep going.
3. Make sure the code repeats the "if" statement at least twice.
4. Find the statements between consecutive "if" statements. These statements will become the body of the loop.
5. The "if" test will become the "while" test.
6. If there's anything before the first "if" test, it will go immediately before the while loop (outside of the body).

---

**Pseudocode for Name Comparison Program**

```python
name1 = ask for first name 
if name1 is not STOP, then keep going 
name2 = ask for second name 
if name1 < name2, print out name1, 
else print name2 
name1 = ask for first name 
if name1 is not STOP, then keep going 
name2 = ask for second name 
if name1 < name2, then keep going 
...blah blah blah
```

```
def main():
    name1 = input("What is name? ")
    while name1 != "STOP":
        name2 = input("What is name? ")
        if name1 < name2:
            print(name1, "is the earlier name in the dictionary")
        else:
            print(name2, "is the earlier name in the dictionary")
    name1 = input("What is name? ")
    print("End of the loop")
main()
```

---

**Generating Random Numbers**

- Random number are useful in a lot of programming tasks
- **random module**: includes library functions for working with random numbers
- **randint function**: generates a random number in the range provided by the arguments

![Generating Random Numbers Diagram](image_url)
Generating Random Numbers

Figure 6.3. The random function returns a value

A random number in the range of 1 through 100 will be assigned to the random variable

Figure 6.4. Displaying a random number

A random number in the range of 1 through 10 will be displayed.

Using Random Numbers Example

```
# This program displays a random number
# in the range of 1 through 10.
import random

def main():
    # Get a random number.
    number = random.randint(1, 10)
    # Display the number.
    print('The number is', number)
    # Call the main function.
    main()
```

Pseudocode for Guess the Number Game

number = generate a random number between 1 and 100
guess = ask user to guess a number between 1 and 100
if number does not equal guess, then keep going
if guess is greater than number, tell user this else if guess is less than number, tell user this guess = ask user to guess a number if number does not equal guess, then keep going if guess is greater than number, tell user this else if guess is less than number, tell user this guess = ask user to guess a number between 0 and 100 if number does not equal guess, then keep going if guess is greater than number, tell user this else if guess is less than number, tell user this . . .
When user guesses correct number, tell them so

Sample Output

User's guess was too high.
Please try again.
Your guess was too low.
Please try again.
You guessed right! Great work.

Loops that Count

```
cnt = 1
while cnt <= 100:
    if cnt % 2 == 0:
        print(cnt, 'is even')
    else:
        print(cnt, 'is odd')
    cnt += 1
```

These loops are equivalent.
Examples of loops that count

• See count1.py in Box.com folder

Practice

1. Write a while loop that prints all divisors of 30.
   — Your code should print out the following:
   1, 2, 3, 5, 6, 10, 15, 30

2. Modify this loop to print out all divisors of 30 AND 50

3. Now let the user select any 2 integers and print out the common divisors of these 2 integers

4. Challenge: Print out only the largest of the common divisors of these 2 numbers