Chapter 5
Repetition Structures
Sections 5.1-5.3
Bonita Sharif

Scenario
• You need to calculate the bonus for several employees.
• There needs to be a way that you can specify the number of times you would like to perform a certain task.
• One way is just to keep duplicating the same code over and over – Why is this bad?

What is a Repetition Structures?
• A repetition structure causes a set of statements to execute repeatedly
• Repetition is also known as looping

Two Types of Loops
• Condition controlled
  – Uses a true/false condition to control the number of times a loop repeats
  – while loop
• Count controlled
  – Repeats a specific number of times
  – for loop

While loops
• Causes a set of statements to repeat as long as the condition is true
  – Pre-test loop
• Loop exits (stop repeating) when condition is false
• Each execution of the loop is called an iteration
• Draw flowchart

while loop syntax
# Python while loop
while condition:  # while clause
  statements

2
3
4
5
6
# commission.py
# This program calculates sales commissions.

def main():
    # Create a variable to control the loop.
    keep_going = 'y'

    # Calculate a series of commissions.
    while keep_going == 'y':
        # Get a salesperson's sales and commission rate.
        sales = input('Enter the amount of sales: ')
        comm_rate = input('Enter the commission rate: ')

        # Calculate the commission.
        commission = sales * comm_rate
        # Display the commission.
        print('The commission is $%.2f.' % commission)

        # See if the user wants to do another one.
        keep_going = raw_input('Do you want to calculate another ' + 'commission (Enter y for yes): ')

    # Call the main function.
    main()

Infinite Loops

- A loop should always have a way to terminate.
- Otherwise the program will never end until you kill it
- An infinite loop is a logic error
- See infinite.py for an example of an infinite loop
- <Ctrl>+C kills the execution

Loop statements

- Statements inside a loop can also include function calls.
- It is good programming practice to use functions inside loop.
  - Improves readability

for loops

- Causes a set of statements to repeat/iterate a specific number of times
- Work on a sequence of data items
- When a for loop executes, it iterates once for each item in the sequence

for loop syntax

```python
# Python for loop
for variable in [value1, value2, etc...]: # for clause
    statements

# target variable
A list in Python is enclosed in [] separated with commas
```

for loop Mechanics

1. The variable (target variable) is assigned the first number in the list.
2. Statements in the block are executed next.
3. The variable is assigned to the next value in the list
4. Jump to step 2 (until no more values exist in the list)
for loop Example

Look at ForLoops.py in the examples folder

range Function

• Using for loops is easier with the range inbuilt function.
• Instead of using a list of values you call the range function with a number or a set of numbers.

Using the range function

• Calling range with 1 argument
  – range(3)
  – This will generate a list of integers in the range of 0 up to 3 (but not including 3)
• Calling range with 2 arguments
  – range(1, 5)
  – This will generate a list of integers from 1 to 5 (but not including 5)

Using the range function

• Calling range with 3 arguments
  – range(1, 20, 3)
  – The last argument is the step value.
  – This will generate a list of integers
    1, 4, 7, 10, 13, 16, 19

Exercises

• What will the following display?
  for number in range(0, 501, 100)
    print number

  for number in range(10, 5, -1)
    print number