15-112 Fundamentals of Programming

Lecture 6

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Announcements

- Assignment 2 is due Tuesday, Sept 8

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Conditional Execution

```
number = int( input("Enter a number "))
if number > 0:
    print ("The number is positive“)
print ("Thank you for your number“)
```

```
>>>
Enter a number 34
The number is positive
Thank you for your number
>>>
```

```
>>>
Enter a number -5
Thank you for your number
>>>
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

More on forming conditions

- Conditional Operators
  - and
  - or
  - not

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Combining conditions

```
num1 = int(input())
num2 = int(input())
num3 = int(input())
if num1  > num2 and num1 > num3:
    print (num1)
if num2  > num1 and num2 > num3:
    print (num2)
if num3  > num1 and num3 > num2:
    print (num3)
```


If else

- Sometimes we need to execute some alternate statement

```
import math
num = int(input("Enter a number "))
if num >= 0:
    print ("Factorial is",math.factorial(num))
else:
    print ("You have entered an invalid number“)
```

```
>>>
Enter a number 5
Factorial is 120
```

```
>>>
Enter a number 0
You have entered an invalid number
```

If-elif-else

- Sometimes we need to make mutually exclusive choices

```
score = int(input("Enter your score "))
if score >= 90:
    print ("You have an A")
if score >= 80:
    print ("You have an B")
if score >= 70:
    print ("You have an C")
if score >= 60:
    print ("You have an D")
if score < 60:
    print ("You have an R")
print ("Now you know your grade")
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

If-elif-else

- Fixed grades

```
score = int(input("Enter your score "))
if score >= 90:
    print ("You have an A")
elif score >= 80:
    print ("You have an B")
elif score >= 70:
    print ("You have an C")
elif score >= 60:
    print ("You have an D")
else:
    print ("You have an R")
print ("Now you know your grade")
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Testing

- Get Grade Function

```
def getGrade( score)
    if score >= 90:
        return “A”
    if score >= 80:
        return “B”
    if score >= 70:
        return “C”
    if score >= 60:
        return “D”
    return “R”
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Testing

- Or

```
def getGrade( score)
    grade = “R”
    if score >= 90:
        grade =  “A”
    elif score >= 80:
        grade = “B”
    elif score >= 70:
        grade =“C”
    elif score >= 60:
        grade = “D”
    return grade
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Testing the grade function

- How do you test this function to make sure it works properly?

```
assert(getGrade(85)== “B”)
assert(getGrade(80)== “B”)
assert(getGrade(95)== “A”)
assert(getGrade(90)== “A”)
assert(getGrade(75)== “C”)
assert(getGrade(79)== “C”)
assert(getGrade(70)== “C”)
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Exercise

- Given two circles (center points and radius), return True if the circles intersect and False if they don’t

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

One more

- **nearestBusStop(street)**
  Write the function nearestBusStop(street) that takes a non-negative int street number, and returns the nearest bus stop to the given street, where buses stop every 8th street, including street 0, and ties go to the lower street, so the nearest bus stop to 12th street is 8th street, and the nearest bus stop to 13 street is 16th street.

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Repetition

- I will think of a number between 1 – 100, you try to guess it
- How would you do this in Python?

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

While loop

While (some condition is true) :

    first statement

    second statement

    ….

Outside of Loop

Inside the loop
(Loop Body)

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Using While loops

- Read grades from the user, until the user enters -1. Print the average of all grades.

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

For Loop

- Normally used when you want to execute some code a fixed or known number of times
- Example: Print a name 10 times

```
name = input("Enter your name")
for x in range(10):
    print (name)
```


Range Function

- What does range function do?
  - range(a,b)
    - Generate a list of numbers from a to b – not including b
  - range(a,b,i)
    - Generate a list of numbers a to b in increments of i
      - range(3,11,2) = [3,5,7,9]
      - range(4,12,5) = [4,9]
      - range(12,4,-2) = [12,10,8,6]

for Loop

for i in range(4,10)

=

for each value of i in [4,5,6,7,8,9]

During the loop:

First iteration	→ i = 4
Second iteration	→ i = 5
Third iteration	→ i = 6
Fourth iteration	→ i = 7
Fifth iteration	→ i = 8
Sixth iteration	→ i = 9

Example

- Print the sum of all numbers from 1 – 10

```
sum = 0
for i in range(1,11):
    sum = sum + i
print (sum)
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Another Example of for loop

- Read a string from the user, print each character of the string on a separate line

```
value = input("Enter a string")
for i in range(0,len(value)):
    print (value[i])
```


Nested Loops

```
for i in range(0,2):
    for j in range(0,3):
        print (i," ",j)
```

What would be the output?

Nested Loops

```
for i in range(0,5):
    for j in range(i,5):
        print (i," ",j)
```

What would be the output?

Nested Loops

```
for i in range(0,5):
    for j in range(0,i):
        print (i," ",j)
```

What would be the output?

Conditionals in loops

```
for i in range(0,20):
    if i % 2 == 1:
        print (i)
```

What would be the output?

Problem solving using loops

- **isPerfectSquare(n)**
  Write a function that takes an integer and returns True if it is a perfect square (the square of another integer) and False otherwise.
- hasConsecutiveDigits(n)
- PrintnthPrime(n)

Problem solving using loops

Write the function nthFibonacciNumber that takes a positive integer n and returns the nth Fibonacci number, so:

- nthFibonacciNumber(1) returns 1
- nthFibonacciNumber(2) returns 1
- nthFibonacciNumber(3) returns 2
- nthFibonacciNumber(4) returns 3
- nthFibonacciNumber(5) returns 5
- nthFibonacciNumber(6) returns 8