

~~Math Day 2~~
Programming:
How to make computers do math for you

Matt Coles

February 10, 2015

1 Intro to Python (15min)

Python is an example of a programming language. There are many programming languages. My first language was Fortran. Another well known language is C. Today though; Python.

We will use this in browser python: <http://repl.it/languages/Python> to write and run code. On the left is where we will write code (this usually takes place in an editor like notepad) On the right we have the ‘command line’ where the outputs will be displayed.

The idea is to work through a series of examples to get familiar with how everything works. After which we will be able to tackle some harder mathematical problems as well as make some fun programs.

Example 1. *The print command tells the computer to print an output to the screen. This is everyones first program.*

```
print "hello world"
```

Example 2. *To make your code more readable you should use comments.*

```
# this is a comment; it will do nothing.  
print "hello world" # prints to the screen
```

Example 3. *Variables. Computers can store things! We can assign letters and numbers to a variable using =.*

```
name = "Matt" #letters and numbers in quotes are called a string
print "Hi", name
```

Example 4. *Now let's make things more interesting. Get to know your computer!*

```
name="Matt"
fav_colour="black"
age="25"
print "Hi there"
print "So", name, "you are", age, "years old."
print "I see your favourite colour is", fav_colour, "... nice!"
```

Example 5. *Inputs! You can also have the computer ask the user for information.*

```
print "hello there" # prints to the screen
print "What is your name?"
name = raw_input(">") #shows prompt and waits for input
print "hello",name, "it's a pleasure to meet you"
#your input became the variable: name
```

2 Numbers (40min)

Example 6. *In order to do any math we're going to need some numbers. Numbers are different from strings.*

```
apples = 3
print apples
oranges = apples + 1
print oranges
```

Example 7. *Let's try assigning some more numbers. Before you run, what do you think will print? Surprised?*

```
apples = 1
oranges = apples
apples = 2
print apples, oranges
```

Example 8. *Swap. Start with*

```
apples = 1
oranges = 2
```

now write a few more lines to swap the values in apples and oranges.

Example 9. *Operations! We have the usual addition, subtraction, multiplication, division and exponents.*

```
x = 5
y = 2
print x + y
print x - y
print x*y
print x/y
print x**y
```

```
x = 5.0
y = 2.0
print x + y
print x - y
print x*y
print x/y
print x**y
```

```
print (x + y)**2 - y**(0.5) + x*(y + 3)
```

Now we are seeing the difference between integers and floating point numbers. The computer treats integers and decimals differently. An integer can be represented exactly in the computers memory while a real number can only be stored approximately. Operations with integers are faster and exact. Sometimes you want integers, sometimes you want floats. For floats remember to put a decimal in your numbers. We can now combine numbers in complicated ways.

To change a float to an integer and an integer to a float try:

```
x = int(5.4)
y = float(3)
print x, y
```

To get the fractional part of an integer division try:

```
x=5
y=2
print x, "divided by", y, "is", x/y, "with remainder", x%y
print "or about", float(x)/float(y)
```

Example 10. *If you want to do inputs with numbers you'll need to tell the computer whether the input is an integer or float.*

```
print "I'm your computer. Let's get to know each other"
print "Please enter your favourite number."
your_fav=int(raw_input(">"))
print "so your favourite number is", your_fav, "eh?"
comp_fav = 7
print "my favourite number is", comp_fav
```

Example 11. *Write a program that asks for a number and then returns the number squared and/or write a program that asks for two numbers and returns their sum and/or write a program involving numbers to do something fun.*

Example 12. Quadratic Formula. *Write a program that asks for values a , b and c and solves for x in $ax^2 + bx + c = 0$. Check your answers with Wolfram Alpha.*

Example 13. *At times today we may want to have Python produce a (sudo)random number. A handy way to do this is to use the 'randint' command. There is a library called 'random' that has lots of commands about random numbers. We need to tell Python that we're going to be using it.*

```
from random import randint
x = randint(0,10) # picks a random number between 0 and 10
print x
```

Example 14. *Try writing a program to use random numbers. Write a program that asks for two numbers and returns a random number between the two inputs.*

3 Conditional Statements (30min)

Example 15. *Now on to if statements. We usually want the computer to do different things depending on the situation.*

```
from random import randint
x = randint(0,10)
#print x

if x > 5:
    print x, "is bigger than 5!"
elif x == 5:
    print x, "is exactly 5!"
else:
    print x, "is smaller than 5!"
```

Example 16. *Another example with words and inputs.*

```
print "Hello, I'm your computer."
print "What is your favourite colour?"
fav = raw_input(">")
comp_fav = "purple"

if fav == comp_fav:
    print "That's my favourite too!"
else:
    print "That's a pretty sweet colour."
```

Example 17. *Write a program that asks you a random addition question. If you're right it congratulates you. If you're off by 1 it comments on your closeness. If you're wrong it is sad.*

Example 18. *Let's try making a choose your own adventure game.*

```
print "You are in a cave."
print "There is a path to your Left and a path to your Right."
path = raw_input(">")

if path == "Left":
    print "You see a bear eating cake."
```

```

print "You can Take Cake or Dance."
bear = raw_input(">")
if bear == "Take Cake":
    print "You have angered the bear. It eats you."
elif bear == "Dance":
    print "You and the bear have a dance party."
else:
    print "The bear doesn't like that. It eats you."
elif path == "Right":
    print "You see a calm pool of water."
else:
    print "You should have picked Left or Right!"

```

Example 19. *Try making your own adventure game and/or any other program with if statements.*

Example 20. *Improve your quadratic formula code to print a nice message (rather than just produce an error) if you get a negative under the square root. Bonus: Have the program find the complex roots.*

4 Break Time! (15min)

5 For Loops/Activities (50 min)

The real power of programming starts to come out with loops. Using loops we can make the computer do a lot of work with just a few lines of code.

Example 21. *Here is the basic 'for' loop. Play around with it to get the idea.*

```

for i in range(0,5):
    print i
print "The loop is now done."

```

Example 22. *Write a program that finds the sum of the first 10 numbers. Add prints to get a feel for what happens at each step. When you're comfortable find the sum of the first 100 numbers.*

```
total=0
for i in range(1,10+1):
    total = total + i
    # print "The counter is", i
    # print "The total is at:", total
print "The whole total is:", total
```

Example 23. *Make python type the numbers from 1 to 10 but skip your favourite number.*

Example 24. *Write a program that prints random numbers until you get your favourite number.*

Example 25. *Root finding by bisection! Last time we looked at the graph of $y = x^3 - 3x^2 + 1$. Write a program to approximate the values where the graph crosses the x -axis. Check your answer with wolfram alpha.*

Example 26. *In the 12 days of Christmas song how many gifts does your true love have to buy in total? Consider using a loop within a loop! Bonus: Try to get your program to print the lyrics of the song.*

Example 27. *Expand on your adventure game (or other kind of game). Try including some math or a loop. Try having a goal and the ability to win. Try a game that asks you math questions until you get 10 right. For this look up the ‘while loop’ and give it a try.*

If you’re looking to learn Python I recommend “Learn Python the Hard Way” <http://learnpythonthehardway.org/book/> It is free to read online and will walk you through things like installation.

If you’re looking for math problems to tackle with programming check out Project Euler: <http://projecteuler.net>