Computer Science 134C
Introduction to Computer Science, in Python
Lecture #2 (Values and expressions)
September 10, 2018

Keywords
comment, expression, float, integer, keyword, precedence, script, string, type, value, variable

We explore the simplest expressions of computation in Python.

1. Things to do:

   (a) If you haven't done so, already, visit Mary (office in TCL312, behind door with access code #646404). to get CS account and password, *before lab*.

   (b) Homework 0 is due today in class.

   (c) TA hours start tonight, with 7-9pm, TCL217a ("The Mac Lab").

   (d) Lab today and tomorrow. Please come to your scheduled lab!

2. Basic concepts:

   (a) Python runs in interactive mode for basic calculation.

   (b) We can collect commands into *scripts* and have python evaluate them collectively.

   (c) Python ignores everything after hash-marks: #. These are *comments*

   (d) *Variables* are named storage locations. You create new variables by assigning them *expressions* of values.

   (e) Some words in Python are keywords; these words have special meaning. Keywords cannot be used as variable names:

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	while
and	del	global	not	with
as	elif	if	or	yield
assert	else	import	pass	
break	except	in	raise	

   (f) Python supports many built-in types. Examples: `int`, `float`, `str`.

   (g) Each type supports a wide variety of operations. Where possible, operations act similarly on different types.

   (h) There are a great variety of *packages* that collection functions and constants that extend python. For example, the `math` package contains trigonometry definitions, the `random` package has procedures for calculating random values, and the `time` package can help with timing things.

3. Playing.

   (a) How do we compute the roots of a quadratic, $ax^2 + bx + c$? (Hint: $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$.)

   (b) How might we calculate the height of our belly button, in feet and inches, given our height in centimeters? (Hint: the Golden Section.)

(c) How do we compute the day of the week (Sunday is 0, Monday is 1, etc.) given the time in seconds since the 1960s? (Hint: We need the `time` function from the `time` package. Also the 1960s ended on a Wednesday.)

(d) Suppose we wanted to compute the day of the week for an arbitrary date, specified using a month, day, and year?

We'll limit ourselves to dates between 1900 and 2099, inclusive.[1] The computed remainder tells us the day of the week, where 0 is Saturday.

If you do this in your head, you need to remember a short table of monthly adjustments. Each entry in the table corresponds to a month, where January is month 1 and December is month 12.

Month	1	2	3	4	5	6	7	8	9	10	11	12
Adjustment	1	4	4	0	2	5	0	3	6	1	4	6

Notice that 144 is 12^2, 025 is 5^2, 036 is 6^2, and 146 is a bit more than 12^2. If the year is divisible by 4 (it's a leap year) and the date is January or February, you must subtract 1 from the adjustment.

Now, here's the algorithm:

1. Write down the date numerically. The date consists of a month between 1 and 12, a day of the month between 1 and 31, and the number of years since 1900. Grace Hopper, computer language pioneer, was born December 9, 1906. That would be represented as year 6. Tally, the Wonder Dog, was born on August 28, 2017, year 117.

2. Compute the sum of the following quantities:
   - the month adjustment from the given table (*e.g.*, 6 for Admiral Hopper)
   - the day of the month
   - the year
   - the whole number of times 4 divides the year (*e.g.*, 29 for Tally)

3. Compute the remainder of the sum of step 2, when divided by 7. The remainder gives the day of the week, where Saturday is 0, Sunday is 1, *etc*. Notice that we can compute the remainders *before* we compute the sum. You may also have to compute the remainder after the sum as well, but if you're doing this in your head, this considerably simplifies the arithmetic.

How might you adjust the algorithm to make the result of 0 indicate Sunday?

$\star$

---

[1] This particular technique is due to John Conway, of Princeton University. Professor Conway answers 10 day of the week problems before gaining access to his computer. His record is well under 15 seconds for 10 correct dates. See "Scientist at Work: John H. Conway; At Home in the Elusive World of Mathematics," *The New York Times*, October 12, 1993.