

Python

Offensive and Defensive Tool Construction

Table of Contents

[Objectives 4](#_Toc30017879)

[Background Reading 4](#_Toc30017880)

[Important Information 4](#_Toc30017881)

[Introduction: Why Python? 5](#_Toc30017882)

[Problem 1 6](#_Toc30017883)

[Problem 2 6](#_Toc30017884)

[Problem 3 6](#_Toc30017885)

[Problem 4 7](#_Toc30017886)

[Problem 5 7](#_Toc30017887)

[Problem 6 8](#_Toc30017888)

[Problem 7 8](#_Toc30017889)

Offensive and Defensive Tool Construction

Python Programming I

Objectives

This lab focuses on the following objectives:

* Explain the purpose of scripting languages and Python.
* Explore the basic syntax of Python and compare it to C.
* Use variables, expressions and statements in Python.
* Explain function definitions and function calls (pure and with return value).

Background Reading

* Read chapters 1–5 in *How to Think Like a Computer Scientist: Learning with Python*, available at [www.greenteapress.com/thinkpython/thinkCSpy.pdf](http://www.greenteapress.com/thinkpython/thinkCSpy.pdf).
* <https://docs.python.org/3.8/>

# Important Information

All scripts must have the following elements:

1. File and Header comments, which follows the following format:

***# Filename: m##XXX.py***

***# Author: Taylor Swift***

***# Course: ITSC203***

***# Details: This program calculates the rate at which users allow themselves # to be hacked.***

***# Resources: https://www.cs.siue.edu/programming-style-guide***

1. Comments on lines where you used some unique computation that might be tricky to comprehend a month later.

***list1 = [x for x in range(20) if x % 4 == 1] # Using list comprehension to ….***

# Introduction: Why Python?

Both C and Python are high-level languages, and both need to be translated into machine code to execute. C is compiled whereas Python is interpreted.

Describe the differences between the two using the table below (draw a simple diagram of the path from source code to program execution and output):

|  |  |
| --- | --- |
| **C Program** | **Python Program** |
|  |  |

1. You have learned how to program in C. Examine this simple program in C:

#include <stdio.h>

void main (int argc, char \*\*argv)

{

printf (“Hello World!\n”);

}

1. Write the equivalent program using Python.

print(“Hello World \n”)

Instructor sign-off: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Problem 1

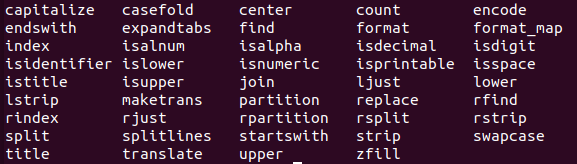
Write a Python program named **m1p1.py** (module 1, problem 1) that outputs a single line consisting of the year, month, date, your SAIT email address, and your first and last name, each separated by a single space. Use the following format (substitute your own name and email for “john smith”). Every time the code runs, the date will be updated to reflect the current date and time.

Jan-17-2020 : 20:37:33 john.smith@edu.sait.ca John Smith

# Problem 2

The image below shows the methods available for the String Class. Write the python script that will generate the following output:

1. The script should be called m1dir.py



# Problem 3

Python has 30+ keywords. Write a Python program named **m1p3.py** that outputs this list of keywords (one per line, sorted alphabetically), and then outputs the same list but sorted alphabetically in reverse.

and

as

…

yield

yield

…

as

and

# Problem 4

The values below are test scores, out of 112, for a course in a far-away galaxy.

You have been tasked to write a program that does the following:

1. Calculate the average of the marks and print that value as a percentage.
2. Calculate the percentage of marks that are above the average print the value as a percentage.
3. Calculate the percentage of marks that are below the average print the value as a percentage.

Hint: Use the format method. Is there a built-in method called sum and len?

**79,66,50,45,45,55,68,88,55,46,54,76,98,77,93,**

**45,87,42,58,81,48,94,52,62,84,79,74,80,80,90,**

**70,65,64,51,43,45,50,66,87,85,40,96,71,82,49,**

**50,56,53,66,79,84,48,79,84,90,60,55,58,65,65,**

**61,78,81,73,53,90,90,96,94,83,85,83,63,66,62,**

**80,68,40,97,40,74,42,50,80,81,63,55,69,57,64,**

**40,50,98,62,79,50,99,42,76,64,42,92,44,54,57,**

**74,44,55,44,44,75,47,57,95,72,92,86,52,80,54,**

**44,100,81,68,68,69,50,79,86,99,78,71,49,76,82,**

**67,51,46,88,87,90,45,89,78,56,41,89,80,46,98,**

**71,91,52,44,66,86,92,67,42,74,65,51,71,53,49,**

**71,75**

# Problem 5

You have been told by your boss to determine which computers are currently on the network. He further tells you that your network is 10.100.16.0/20; and wants you to generate a list of IP Addresses to scan. He itemizes what he believes you should do as follows:

1. Create a program called **m1subnet.py**
2. Write code that will generate a list of all possible IP Address (including the subnet network address and subnet broadcast address).
3. The output should have the following format:

Subnet Network Address: 10.100.16.0

Subnet First Address: 10.100.XX.XX

Subnet Last Address: 10.100.XX.XX

Subnet Broadcast Address 10.100.XX.XX

Range of IP Address to be scanned: 10.100.XX.XX, 10.100.XX.XX, 10.100.XX.XX, … and so on.

# Problem 6

Write a python program that does the following task:

1. Create a python script with the following name **m1builtins.py**
2. Print a list of all the built-in functions.
3. Each line of output should be no longer than 40 characters
   1. Lines can be shorter than 40
4. Ensure that the names starting with uppercase letters come after the lowercased names.
5. The names should retain case, in the final output.
6. **HINT**: There is a module called \_\_builtins\_\_ containing all builtin functions in python

Example:

all lowercase words yearly

Before UpperCased Words

# Problem 7

Write a Python program named **m1p7.py** that prints a list of all users on your Linux system who have a home directory in the format **/home/username**, and print a pretty table, containing the username using 20 characters and aligned left, and the home directory.

You have been given the following list:

**[['roger', '/home/users/roger'],[ 'edgar', '/home/edgar'],[ 'romeo', '/home/users/romeo'],[ 'tito', '/home/hut/tito'],[ 'reese', '/home/users/reese']]**

commandprompt$ m1p7.py

+-----------+---------------------------+

| roger | /home/users/roger |

| tito | /home/hut/tito |

+-----------+---------------------------+