|  |  |  |
| --- | --- | --- |
| **College of Engineering**  Computer Science & Eng. Dept.  **Course:** CMP 321L  Programming languages Lab | A picture containing logo  Description automatically generated | **Course Professor:** Dr. Michel Pasquier  **Lab Instructor:** Praveena Kolli  **Office:** EB2-126  **Phone**: 971-6-5152352  **e-mail**: pkolli@aus.edu  **Semester**: Summer 2022 |

**Lab 1 – Python String functions**

**Objectives:**

* Understand and apply slicing lists and strings
* Get practice with string and list processing
* Manipulate text using string functions
* Format output with string format and F-strings

**Due date: End of the lab. (**Only one team member needs to submit.)

**Rules:**

(1) Usage: **You should explore and make good use of the Python features you learned in class.** (2) Scope: **You should only use those features that have been explained in detail in class.**

(3) Style: Follow standard Python programming style and conventions.

(4) Logic: Add appropriate comments to your code to explain your solution.

(Code / answers that do not follow the above specifications will be penalized.)

***Warning:* You need to come to the lab properly prepared i.e.**

(1) Make sure you have studied and understood the class material.

(2) Read the lab doc, think about the problems, and prepare questions as needed.

If you do not, completing the lab in 2.45 hours may become too much of a challenge!

**Useful resources:**

* https://docs.python.org/3/tutorial/datastructures.html

**Lab By:**

Rohan Mitra 85023

Keshav Ramesh 88595

**Github link for code**: https://github.com/ro1406/CMP321Lab

**Exercise 1: Using lists and other Python features [1 Marks]**

Write a Python script that defines a list of numbers (50, 12, 27, 33, 61, 49, 28), applies Bubble Sort to it, and prints the result. To do so, you must convert the following C++ code to *proper* Python code, using the language’s features and simplifying as much as possible.

#include <iostream>

using namespace std;

int main()

{

int values[] = {50, 12, 27, 33, 61, 49, 28};

int size=7;

int temp;

for (int i=1; i<size; ++i)

{

for (int j=0; j<size-i; ++j)

if (values[j] > values[j+1])

{

temp = values[j];

values[j] = values[j+1];

values[j+1] = temp;

}

}

cout << "Array after bubble sort:";

for (int i=0; i<size; ++i)

cout << " " << values[i];

}

**Answer:**

values = [50, 12, 27, 33, 61, 49, 28]

for \_ in values:

for j in range (len(values)-1):

if values[j]>values[j+1]:

values[j], values[j+1] = values [j+1], values[j]

print("Array after bubble sort: ", end = ""); print(values)

**Screenshot:**

****

**Exercise 2: Slicing strings and lists [2 Marks]**

1. Write a Python script that defines the string "Welcome to Python" and performs the following, where each operation must consist of one line of Python code using *slicing* only:

* Prints all characters from the 6th to the 10th character.
* Prints all characters after the 4th character.
* Prints the last five characters (without assuming the size).
* Checks and prints whether a given string s is a palindrome (same in reverse).

**Answer:**

string = "Welcome to Python"

print(string[5:10]) #6th char at index 5, 10th char at index 9

print(string[4:]) #5th char onwards (AFTER 4th char)

print(string[-5:])

print(string == string[::-1])

1. Write a python program/script that creates a list that contains the following countries and capitals, in order: "Cairo", "Egypt", "Baghdad", "Iraq", "Delhi", "India", "Tehran", "Iran", "Riyadh", "Saudi Arabia", "Ankara", "Turkey"; and performs the following:

* Prints all cities i.e., the elements at even indices.
* Prints all countries i.e., the elements at odd indices, in reverse.
* Finds and replaces elements Delhi and India with Muscat and Oman.

**Answer:**

places = ["Cairo", "Egypt", "Baghdad", "Iraq", "Delhi", "India", "Tehran", "Iran", "Riyadh", "Saudi Arabia", "Ankara", "Turkey"]

print(places[::2]) #Even indicies

print(places[::-2]) #Odd indicies backwards

places[places.index('Delhi')]="Muscat"

places[places.index('India')]="Oman"

print(places)

**Screenshot:**

Text

Description automatically generated

**Exercise 3: Using string functions, split/join [2 Marks]**

1. Given a list of full names, print acronyms for each by combining the first letter of the first name and the last name (ignoring middle names, if any). Example:

nameList = ["Abdulla mhd zayed", "rashid asif", "john elton rowan smith", "Ali Rami"]

printed acronyms: AZ, RA, JS, AR.

1. Given a string, such as "Welcome to UAE. uae is awesome, right?", find all occurrences of "UAE" in that string (ignoring case).

**Answer:**

#part a

nameList=["Abdulla mhd zayed", "rashid asif", "john elton rowan smith", "Ali Rami"]

for fullName in nameList:

print(fullName.split()[0][0].upper()+ fullName.split()[-1][0].upper(),end= " ")

print()

#part b

s="Welcome to UAE. uae is awesome, right?"

print(s.lower().count('uae')) #.lower to ignore the case

**Screenshot:**

Text

Description automatically generated

**Exercise 4: Using string functions, split/join [3 Marks]**

1. Write a function that prints any given integer, such as 314159265358979, in the following (readable) format:

3\_141\_592\_653\_589\_793

**Answer:**

def addUnderScores(x):

x=str(x)

ans=list()

while(x):

ans.append(x[-3:])

x = x[:-3]

return ('\_'.join(ans[::-1]))

print(addUnderScores(3141592653589793))

1. Write a function that takes as parameters a phone number and a (country-specific) phone number format and returns a string that represents the formatted phone number.

Examples:

UAE: phoneNumber = 971506455734, phoneFormat = "3+2+3+4"

formatted output: 971-50-645-5734

France: phoneNumber = 33109758351, phoneFormat = "2+1+2+2+2+2"

formatted output: 33-1-09-75-83-51

India: phoneNumber = 918966428361, phoneFormat = "2+3+7"

formatted output: 91-896-6428361

Note: Use only string functions not regular expressions.

**Answer:**

print('-'\*95)

def splitNum(num,form):

num=str(num)

arr=list(map(int,form.split('+')))

ans=""

currIndex=0

for x in arr:

ans+=num[currIndex:currIndex+x]+"-"

currIndex+=x

return ans[:-1]

print(splitNum(num = 971506455734, form = "3+2+3+4"))

print(splitNum(num = 33109758351, form = "2+1+2+2+2+2"))

print(splitNum(num = 918966428361, form = "2+3+7"))

**Screenshots:**

**Text

Description automatically generated**

**Exercise 5: string format and F-strings [2 Marks]**

1. Display sin and cos values of n in the range 1 to 10 in a tabular format as given below with 5 decimals. Use formatting method from string class.

﻿1 0.84147 0.54030

2 0.90930 -0.41615

3 0.14112 -0.98999

4 -0.75680 -0.65364

5 -0.95892 0.28366

6 -0.27942 0.96017

7 0.65699 0.75390

8 0.98936 -0.14550

9 0.41212 -0.91113

10 -0.54402 -0.83907

1. Using F-String format print n and 2\*\*n as given below. n is in the range 0 to 160 increment by 8. n is right aligned, and 2\*\*n is center aligned.

**Graphical user interface, application

Description automatically generated**

**﻿**

**Answer:**

**#**part a

import math

for i in range(1,10+1): print('{:2d} {:10.5f} {:10.5f}'.format(i, math.sin(i), math.cos(i)))

print("-"\*95)

#part b

for n in range(0,160+1,8): print(f'{n:3d} {2\*\*n:^49}')

**Screenshot:**

**Graphical user interface, text

Description automatically generated**

**Text

Description automatically generated**