1811ICT/2807ICT/7001ICT Programming Principles Workshop 6

School of Information and Communication Technology

Griffith University

|  |  |
| --- | --- |
| *Goals* | This workshop focusses on lists, indexing, slices, list methods, and/or tuples. |
| When | Workshops from Friday 29 April to Thursday 5 May |
| Marks | 3 |
| Due | Pre-workshop questions before the start of the above mentioned workshops  Workshop programming problems by 11:59pm on 8 May |

# Before your workshop class:

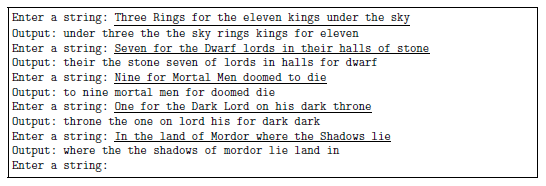
* Read all of this document.
* Review the lecture notes sections 1 to 17.
* **Complete the pre-workshop questions (1 mark) posted on the course website and submit the answers for marking**.

# Workshop activities

At any stage, when you are stuck, *ask your tutor*!

## Problem 1

*Problem:* Write a program that reads strings (without digits or symbols in the string) typed by the user until an empty string is entered. For each string, convert all words to lowercase, then sort and print the words into descending order lexicographically. Hint: use split function to split a string into a list, then sort it with a method.



*Answer*: Copy your code in the space given below and insert screenshots of your program output for the following two scenarios:

* Enter a string: The Old Woman Who Lived in a Shoe
* Enter a string: Birds of Prey and the Fantabulous Emancipation of One Harley Quinn

***Copy your code here***

# Matthew Prendergast

# 2nd May, 2022 - Problem 1 (Workshop - Week 6)

# Prompt user to input a string.

str = input("Enter a string: ")

# Loop until an empty sting has been entered.

while len(str) > 0:

    # Convert the string to lowercase, split it into a list and sort it.

    str = str.lower()

    split\_list = str.split(" ")

    split\_list.sort(reverse=True)

    # Print the new sentence.

    print("Output: ", end="")

    for j in range(len(split\_list)):

        print(split\_list[j], end=" ")

    print()

    # Prompt user to re-enter the string.

    str = input("Enter a string: ")

***Insert your screenshots here***

Text

Description automatically generated

Text

Description automatically generated

## Problem 2

*Problem:* Write a program that allows the user to enter two lists of integers, calculate the sum of the first and the last integers in each list, and print the larger sum. In the event of a tie, print Same. When there is only one integer in the list, the sum is the integer itself.

List 1: 1 2 3 4 5

List 2: 5 6 7

Output: 12

List 1: 4 3 10 1

List 2: 9

Output: 9

List 1: 4 3 2 1

List 2: 1 2 3 4

Output: Same

*Answer*: Copy your code in the space given below and insert screenshots of your program output for the following two scenarios:

* List 1: [2, 4, 5, 7, 8, 9]; List 2: [6, 5]
* List 1: [7, 3, 23, 5, 12]; List 2: [11, 23, 4, 1, 3]

***Copy your code here***

# Matthew Prendergast

# 2nd May, 2022 - Problem 2 (Workshop - Week 6)

# Prompt user to input a a list of numbers.

input\_one = input("List 1: ")

input\_two = input("List 2: ")

# Convert the input to a list.

list\_one = input\_one.split(" ")

list\_two = input\_two.split(" ")

# Count the first and last digit in each list.

if len(list\_one) > 1:

    count\_one = int(list\_one[0]) + int(list\_one[-1])

else:

    count\_one = int(list\_one[0])

if len(list\_two) > 1:

    count\_two = int(list\_two[0]) + int(list\_two[-1])

else:

    count\_two = int(list\_two[0])

# Print the output.

if count\_one > count\_two:

    print("Output:", count\_one)

elif count\_two > count\_one:

    print("Output:", count\_two)

else:

    print("Output: Same")

***Insert your screenshots here***

Text

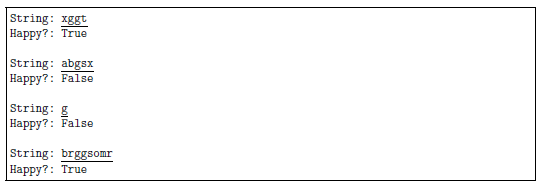
Description automatically generated

Text

Description automatically generated

## Problem 3

*Problem:* We’ll say that a lowercase ’g’ in a string is ”happy” if there is another ’g’ immediately to its left or right. Write a function to print True if all the g’s in the given string are happy, otherwise, print False.

**

*Answer*: Copy your code in the space given below and insert screenshots of your program output for the following two scenarios:

* String: poggpoggpogg
* String: daggergoo

***Copy your code here***

# Matthew Prendergast

# 3rd May, 2022 - Problem 3 (Workshop - Week 6)

# Prompt user to input a string of letters.

str = input("String: ")

happy = "False"

# If the length of the string is more than 1 character, and G is in the string.

if "g" in str and len(str) > 1:

    # Compare every charter, except the last, to the next charter, and the previous character in the string.

    for i in range(len(str) - 1):

        if str[i] == "g":

            if str[i + 1] == "g" or str[i - 1] == "g":

                happy = "True"

            else:

                happy = "False"

                break

    # Check the last character before finishing.

    if str[-1] == "g" and str[-2] != "g":

        happy = "False"

# Print the result.

print("Happy?", happy)

***Insert your screenshots here***

Text

Description automatically generated

Text

Description automatically generated

## Problem 4

*Problem:* Write a program that allows the user to enter the marks of 5 students in 4 courses, and outputs the highest average marks for students and courses. Hint: Consider 2 dimensional lists, i.e. the element of a list is a list.

Student 1 (courses 1-4): 50 60 70 60

Student 2 (courses 1-4): 100 90 87 90

Student 3 (courses 1-4): 70 100 90 90

Student 4 (courses 1-4): 30 65 50 50

Student 5 (courses 1-4): 58 50 74 43

The highest average mark of students: 91.75

The highest average mark of courses: 74.2

*Answer*: Copy your code in the space given below and insert screenshots of your program output for the following two scenarios:

* Student 1 (courses 1-4): 85 70 65 60

Student 2 (courses 1-4): 50 55 75 90

Student 3 (courses 1-4): 67 87 77 55

Student 4 (courses 1-4): 97 88 86 90

Student 5 (courses 1-4): 35 50 55 60

* Student 1 (courses 1-4): 58 77 68 58

Student 2 (courses 1-4): 49 67 89 78

Student 3 (courses 1-4): 56 86 76 66

Student 4 (courses 1-4): 58 88 83 78

Student 5 (courses 1-4): 76 69 77 80

***Copy your code here***

# Matthew Prendergast

# 3rd May, 2022 - Problem 4 (Workshop - Week 6)

# Prompt user to input a a list of students and scores.

input\_one = input("Student 1 (courses 1-4): ")

input\_two = input("Student 2 (courses 1-4): ")

input\_three = input("Student 3 (courses 1-4): ")

input\_four = input("Student 4 (courses 1-4): ")

input\_five = input("Student 5 (courses 1-4): ")

# Append the scores into a 2D list of students.

students = []

students.append(input\_one.split(" "))

students.append(input\_two.split(" "))

students.append(input\_three.split(" "))

students.append(input\_four.split(" "))

students.append(input\_five.split(" "))

# Convert the list to integers.

for i in range(len(students)):

    for j in range(len(students[i])):

        students[i][j] = int(students[i][j])

# Initialise the variables for the highest scores.

high\_student, high\_course= 0, 0

# Find the highest student score.

for i in range(5):

    if sum(students[i]) / len(students[i]) > high\_student:

        high\_student = sum(students[i]) / len(students[i])

# Find the highest course score.

for i in range(4):

    count = 0

    for j in range(5):

        count = count + students[j][i]

    if count / 5 > high\_course:

        high\_course = count / 5

print("The highest average mark of students:", high\_student)

print("The highest average mark of courses:", high\_course)

***Insert your screenshots here***

A screenshot of a computer

Description automatically generated

Text

Description automatically generated

# Submission and marking

The pre-workshop can be accessed and submitted online using the provided link in the course website. Students get 1 mark if they get >50% in pre-workshop questions, or 0.5 mark if they get 0%-50% in pre-workshop questions, or 0 marks without any attempt.

For workshop tasks, please submit this document with copied codes and inserted screenshots using the provided submission link in the course website. Students get 2 marks if they complete three or more problems correctly, or 1 mark if they complete one or two problems correctly, or 0 marks without any attempt.