1811ICT/2807ICT/7001ICT Programming Principles Workshop 4

School of Information and Communication Technology

Griffith University

|  |  |
| --- | --- |
| Goals | In this workshop we create interactive scripts that make decisions and/or loop. |
| When | Workshops from Friday 8 April to Thursday 21 April |
| Marks | 3 |
| Due | Pre-workshop questions before the start of the above mentioned workshops  Workshop programming problems by 11:59pm on 24 April |

# Before your workshop class:

* Read all of this document.
* Review the lecture notes sections 1 to 13.
* **Complete the pre-workshop questions (1 mark) posted on the course website and submit the answers for marking**.

# Workshop activities (2 marks)

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

## Problem 1

*Problem:* Write a program that reads whole numbers typed by the user until a zero is entered, then prints the number of positive numbers that were entered. Sample run:

Enter a number: 3

Enter a number: -2

Enter a number: 5

Enter a number: 6

Enter a number: -100

Enter a number: 70

Enter a number: 22

Enter a number: 68

Enter a number: 0   
6 positive numbers were entered.

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

***Copy your code here***

# Matthew Prendergast

# 19th April, 2022 - Problem 1 (Workshop - Week 4)

# Retrieve the user input.

number = int(input("Enter a number: "))

# Declare a count variable.

count = 0

# Repeat the question until a user types 0.

while number != 0:

    if number > 0:

        count += 1

    number = int(input("Enter a number: "))

# Print the result.

print(f"{count} positive numbers were entered.")

***Insert your screenshots here***



Text

Description automatically generated

## Problem 2

*Problem:* In mathematics, the Fibonacci sequence is defined such that each Fibonacci number is the sum of the two preceding ones, starting from 0 and 1. That is, F1 = 0, F2 = 1, F3 = 1, F4 = 2, ..., Fn = F(n-1) + F(n-2). Write a program that given an input n, outputs the first n Fibonacci numbers. The format of output is that at most 4 numbers can be displayed in a row. Sample run:

|  |
| --- |
| Enter a positive number: 6  0 1 1 2  3 5  Enter a positive number: 10  0 1 1 2  3 5 8 13  21 34 |

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

* Enter a positive number: 8
* Enter a positive number: 15

***Copy your code here***

# Matthew Prendergast

# 19th April, 2022 - Problem 2 (Workshop - Week 4)

# Retrieve the user input.

number = int(input("Enter a positive number: "))

# Print value if the number is 0.

if number == 0:

    print(0)

# Print value if the number is 1.

elif number == 1:

    print(0, 1)

# Print value for all other number inputs, formatted as required.

else:

    first = 0

    second = 1

    count = 2

    print(first, second, end=" ")

    for i in range(2, number):

        print(first + second, end=" ")

        temp\_first = first

        temp\_second = second

        first = second

        second = temp\_first + temp\_second

        count += 1

        if count == 4:

            print("")

            count = 0

***Insert your screenshots here***

A screenshot of a computer

Description automatically generated

Text

Description automatically generated

## Problem 3

*Problem:* Given an input number n, print a diamond shape with 2\*n-1 rows.

Sample run:

|  |
| --- |
| Enter a positive number: 3  xxx  xxxx  xxxxx  xxxx  xxx |

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

* Enter a positive number: 1
* Enter a positive number: 5

# Matthew Prendergast

# 19th April, 2022 - Problem 3 (Workshop - Week 4)

# Retrieve the user input.

number = int(input("Enter a positive number: "))

# Initilise the variable for the row count and number of spaces and crosses are required per row.

rows = 2 \* number - 1

space, cross = 0, 0

# Build the top half of the diamond.

for i in range(1, rows + 1, 2):

    cross = i

    space = (rows - cross) // 2

    print(" " \* space, "x" \* cross, " " \* space, sep="")

# Build the bottom half of the diamond.

for j in range(rows - 2, -1, -2):

    cross = j

    space = (rows - cross) // 2

    print(" " \* space, "x" \* cross, " " \* space, sep="")

*Text

Description automatically generated*

***Text

Description automatically generated***

## Problem 4 (Optional, 1811ICT students are strongly encouraged to try)

*Problem:* A palindrome is a number or a text phrase that reads the same backwards as well as forwards. Examples of palindromes are 123321, 1234321, 55555, 22, 454, 1, 0. Write a program that reads in a positive integer number, and prints out whether or not that number is a palindrome. Sample run:

Enter a positive number: 12321

12321 is a palindrome

Enter a positive number: 1234

1234 is not a palindrome

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

* Enter a positive number: 345543
* Enter a positive number: 92321

# Matthew Prendergast

# 19th April, 2022 - Problem 4 (Workshop - Week 4)

# Retrieve the user input.

number = input("Enter a positive number: ")

# Initialise the variables to use.

length = len(number)

count = 0

# Create the loop to test the string and print the result.

for i in range(length):

    if number[i] != number[length - (i + 1)]:

        count +=1

if count == 0:

    print(number, "is a palindrome")

else:

    print(number, "is not a palindrome")

Text

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 two or more problems correctly, or 1 mark if they complete one problem correctly, or 0 marks without any attempt.