

2807/7001ICT Programming Principles (I), Trimester 3, 2018

Workshop 8

School of Information and Communication Technology
Griffith University

November 9, 2018

<i>Module</i>	3
<i>When</i>	Day 8
<i>Goals</i>	This workshop is more practice problems with functions, and lists.
<i>Marks</i>	5
<i>Due</i>	Problem 1 by the end of this workshop; problems 2 to 5 by the <i>beginning</i> of the next workshop.

1 Preparation

Before your workshop class:

- Read all of this document.
- Review the lecture notes sections 1 to 21.
- Bring some paper (a print-out of this document is best) and writing implements.
- Bring a storage device, such as a portable hard drive and cable, or a USB drive.

2 Pre-workshop questions

There are no pre-workshop questions for this workshop.

3 Workshop activities

3.1 Marking last workshop's problems

If you have problems that still need marking from the previous workshop, get them marked at the *start* of this one.

3.2 Problem 1 (1 mark)

Problem: Write a *function* that given a list, returns *both* the minimum and maximum values in the list. Your main program should allow the user to enter space-separated numbers as test lists, until a blank line is entered.

```
List: 3 6 9 2 4 6
min = 2, max = 9
List: 7
min = 7, max = 7
List:
```

3.3 Problem 2 (1 mark)

This problem and the rest are due at the beginning of the next workshop. *Ask any questions you have about the problems NOW!*

Problem: Write a *function* that given two lists, returns the list of all the elements in the first list that do not occur in the second list. Your main program will allow the enter two lists of numbers and end input with a blank line for list 1.

```
List 1: 1 3 4 2 1 2 1 3
List 2: 1 1 3
[4, 2, 2]
List 1: 1 3 4 2 1 2 1 3
List 2: 1 2 3 4
[]
List 1:
```

Hint: In your function, do not modify the list parameters as this will change the global lists! When parameters are mutable, changes to the parameters always change the globals!

3.4 Problem 3 (1 mark)

Problem: Write a *function* that given two lists, returns the list of all the elements that occur multiple times in both lists. The returned list should be in ascending order, without duplicates. Your main program will allow the enter two lists of numbers on one line separated by a semicolon and end input with a blank line.

```
Lists: 1 3 4 2 1 2 1 3; 4 4 2 4 3 2 4 4 3 1 3
[2, 3]
Lists : 1 1 2 3 4 5; 2 3 4 5 6
[]
Lists : ;
[]
Lists:
```

3.5 Problem 4 (1 mark)

Problem: Write a *function* that given a list of numbers, returns **True** if and only if all of the numbers in the list form an arithmetic progression, that is the difference between any two successive numbers in the list is the same. Your main program should allow the user to enter space-separated numbers as test lists, until a blank line is entered.

```
List: 1 2 3
True
List: 1 3 4
False
List: 3 2 1 0
True
List: 7
True
List: 7 7
True
List: 10 9 8 7 4 3 2 1
False
List:
```

3.6 Problem 5 (1 mark)

Problem: Write a *function* that given a list of numbers, returns the length of the longest sequence of numbers within that list that forms an arithmetic progression. Your main program should allow the user to enter space-separated numbers as test lists, until a blank line is entered.

```
List: 1 2 3
3
List: 1 3 4
2
List: 3 2 1 0
4
List: 7
1
List: 7 7
2
List: 10 9 8 7 4 3 2 1
4
List:
```

4 After the workshop

- You have created programs that might be useful to refer back to in future workshops. Make sure that you will have that work in the future. One copy is not enough for an IT professional. You should have at least 2 copies:
 1. on your Griffith network storage drive; and
 2. on your portable storage device.
- Please answer these feedback questions.
 - What was the best aspect of this workshop?
 - What was the most difficult or worst aspect of this workshop?
 - Did you find an error in these workshop notes?