German University in Cairo
Faculty of Media Engineering and Technology
Prof. Dr. Slim Abdennadher
Dr. Aysha ElSafty

Introduction to Computer Science, Winter Semester 2016
Practice Assignment 5
Discussion: 5.11.2016 - 10.11.2016

Exercise 5-1

To Be Discussed

Given a list `A` of floating-point numbers, representing students' GPAs, and a bonus mark as inputs from the user, write a Python algorithm that adds the bonus mark to all students' GPAs.

Exercise 5-2

To Be Discussed

The simplest algorithm to search a list of Numbers `N` for a given key Key is to test successively each element.

```
N = eval(input("Enter a list of numbers:"))
m = len(N)
Key = eval(input("Enter a key:"))
i = 0
FOUND = False
while i < m and FOUND == False:
␣ if Key == N[i]:
␣␣ FOUND = True
␣ else:
␣␣ i = i+1
if FOUND == False:
␣ print("Sorry, key is not in the list")
else:
␣ print("Key found")
```

If a list is already stored in increasing order, a modified sequential search algorithm can be used that compares aganist each element in turn, stopping if a list element exceeds the target value.
Write an algorithm for the modified sequential serach.

Exercise 5-3

Given two lists `A` and `B`, write an algorithm that uses looping to store the sum of the corresponding elements of the lists `A` and `B` in a new list `C`.

Exercise 5-4

Dice Role

Write an algorithm that prints a list of n dice six-sided rolls.

Exercise 5-5

Write an algorithm to find the maximum value stored in an (unsorted) list `A`.

Exercise 5-6 Thousand Numbers

Given a list of non-negative numbers. Write an algorithm to find the number of

- even positive numbers
- odd positive numbers
- Zeros

Additionally, the algorithm should find the sum of

- even positive numbers
- odd positive numbers

Exercise 5-7 Print Repeated

Write an algorithm that given an **ordered** list of integers `A` prints the elements in the list that are repeated. If some elements occur more than twice, then these elements should be printed only once.

For example, for the list

```
1 1 1 1 4 6 7 7 8
```

your algorithm should print

```
1 7
```

Exercise 5-8 Reverse List

Write an algorithm that reverses the order of elements of the given list.

Exercise 5-9

Write an algorithm that given a list of integers `A` moves all even elements in a list of integers to the front of the list and all odd elements to the rear. Hint: you do not have to maintain any order other than all evens appearing before all odds in the list. For example: if the list is of the form [1,4,5,6,2,10] then the algorithm should create a new list of the form [4,6,2,10,5,1] and prints the elements of the resulting list.

Exercise 5-10

Write an algorithm that given a list of integers `A` and a number `x` prints the number of occurrences of `x` in the list. In addition, the algorithm should print the positions where x occurs.
For example, if the list is [1, 2, 4, 1, 3] and `x` is 1 then the algorithm should print
1 occurs in the following positions: 0, 3
The number of occurences of 1 is 2
If the list is [1, 2, 4, 1, 3] and `x` is 0 then the algorithm should print
The number of occurences of 0 is 0

Exercise 5-11 String Manipulation - Palindrome

Write an algorithm that determines whether the String the user inputs is a `palindrome` or not. A palindrome is a piece of text that can be read the same way in either direction (left to right and right to left). Examples of palindromes include words such as `racecar` and `noon`.

Exercise 5-12 String Manipulation - Run Length

Given a String containing uppercase characters (A-Z), write an algorithm that compresses repeated ‘runs’ of the same character by storing the length of that run.

```
Example:

Input: WWWWWWWWWWWWBWWWWWWWWWWWWBBBWWWWWWWWWWWWWWWWWWWWWWWWBWWWWWWWWWWWWWW
Output: 12W1B12W3B24W1B14W
```