

## COURSE OF COMPUTER SCIENCE *LABORATORY PRACTICE n. 6*

### **Exercise 1 (1 Point):**

Write a Python program for:

- Loading an array  $v$  of  $DIM$  ( $DIM$  is a constant) integer numbers.
- Copying all its positive elements into an array named  $v_p$  and all its negative ones into another array named  $v_n$ .
- Displaying the contents of  $v_p$  and  $v_n$  (only the values copied from  $v$ ).

**Example:** let  $DIM = 10$  and assume that the following array has been introduced:

```
v = [18 11 -4 5 0 0 -2 3 25 0]
```

Then, the following two arrays must be generated and displayed:

```
v_p = [18 11 5 3 25]
v_n = [-4 -2]
```

### **Exercise 2 (2 Points):**

Write a Python program which:

- Reads from the keyboard a sequence of positive numbers, storing into an array. The sequence is terminated by the introduction of a negative value.
- Removes from this array all the replicated values, by shifting all the remaining elements to the left, i.e., by “compacting” the array to the positions characterized by smaller indices.
- Displays the resulting array (only the “saved” elements).

Assume that the length of the sequence is limited to at most 200 values.

**Example:** let the sequence of numbers introduced by the user be the following:

```
0 15 15 5 0 10 25 5 0 0 5 15 10 40 10 30 -2
```

Then, the following sequence of values must be generated and printed out:

```
0 15 5 10 25 40 30
```

### **Exercise 3 (2 Points):**

Write down a Python program in order to:

- Read an array  $v$  of  $N$  integer numbers ( $N$  is a predefined constant).
- Find *all* the sub-arrays of  $v$  (if any) such that, for each of them, the sum of its elements is equal to 0.

**Example:** let  $N = 11$  and assume that the following array  $v$  has been introduced:

```
v = [2 -3 1 2 3 1 4 -6 7 -5 -1]
```

Then, the program must produce the following output:

```
Sub-array starting from index 0, length 3.
Sub-array starting from index 1, length 3.
Sub-array starting from index 5, length 6.
Sub-array starting from index 6, length 4.
```

#### Exercise 4 (2 Points):

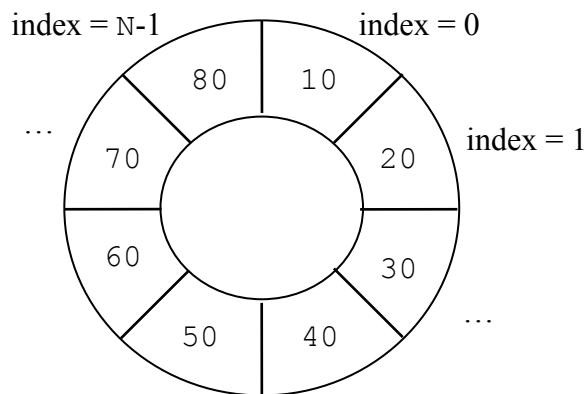
Write down a Python program in order to:

- Load an array  $v$  of  $N$  integer values ( $N$  is a predefined constant).
- Enumerate all the triplets of consecutive values in the array, *considering the last element of the array as glued to the first one* (note that the number of such triplets is exactly equal to  $N$ ). For each triplet, the sum of its elements must be computed and displayed on screen (the order in which the triplets are considered is not relevant).

**Example:** let  $N = 8$  and assume that the following array has been introduced:

$v = [ 10 \quad 20 \quad 30 \quad 40 \quad 50 \quad 60 \quad 70 \quad 80 ]$

Then, the array must be thought as the following figure shows.



Therefore, the program should output a set of messages like:

```
Triplet 1: 10 + 20 + 30 = 60
Triplet 2: 20 + 30 + 40 = 90
Triplet 3: 30 + 40 + 50 = 120
Triplet 4: 40 + 50 + 60 = 150
Triplet 5: 50 + 60 + 70 = 180
Triplet 6: 60 + 70 + 80 = 210
Triplet 7: 70 + 80 + 10 = 160
Triplet 8: 80 + 10 + 20 = 110
```

#### Exercise 5 (2 Points):

Write a Python program which:

- Reads a sequence of integer values, terminating this operation when a value smaller than 0 or larger than 99 is entered.
- Evaluates the number of such values which fall in every range of the following list: [0-9], [10-19], ..., [80-89], [90-99].
- Draw a *vertical* histogram representing these numbers, as shown in the example below.

**Example:** let 1 2 3 4 5 10 20 24 55 62 73 74 89 93 94 95 -1 be the sequence of values introduced.

Then, the following “figure” must be obtained:

```
###  #####
#  #      #  #
#          #
#
#
```