04JCJLZ - COMPUTER SCIENCES - 2015/2016

Laboratory 7

Objectives:

- Write programs that use characters and strings

Technical content:

- Advanced use of functions and arrays
- Using variable's type *char*
- Using functionalities included in *ctype.h* and *math.h*

Preferably to be solved in the laboratory:

Exercise 1. Write a C program that, given two arrays named `vbase` and `vexponent`, both composed of *N* elements, performs the raising of each element in the `vbase` array to the power exponent given by the related element in the `vexponent` array (thus, `vbase[i]` power to `vexponent[i]`). Each result has to be stored into the corresponding position of a third array, named `vres` (thus, `vres[i]`). Use the function *power* defined in the previous laboratory exercise, having the following prototype:

```
int power(int base, int exponent);
```

At first insert the values of the N bases, and then the N exponents; at the end of the program, print the values stored into `vres`.

Example: the user inserts the following values (with N equal to 5):

```
vbase    → 5 2 7 4 9
vexponent → 2 6 1 8 3
```

The resulting array will be the following:

```
vres     → 25 64 7 65536 729
```

Hint: invoke the *power* function as many times as the number of elements of each array and time by time store the result into the appropriate position of the `vres` array.

Exercise 2. Write a C program that:

a. **In the main**, asks the user to insert N values, stores them into the `vect` array, and then asks for a further value: `x`
b. **Passes** the `vect` array and the `x` value **to a function** that performs the multiplication of each array's element by `x`. The prototype is the following:

```
void mult(int v[], int n, int x);
```

The **main** has to print the array after the multiplication.

Note: the function receives the array by reference, thus is able to modify the values of its elements.

Exercise 3. Write a C program that acquires from keyboard a sequence of characters until a "new line" is typed. After this event, the program has to show the following statistics:

a. the total number of characters;
b. the number of alphabetic characters;
c. the number of uppercase characters;
d. the number of digits;
e. the number of spacing characters;
f. the number of words, where a "word" is a sequence of contiguous alphabetic characters (e.g., "hello 123 world !" contains 2 words).

Hint: Use the standard library's functions defined in the `<ctype.h>` header file and perform the acquisition using a single *char* variable.

To be solved at home

Exercise 4. Write a C program that is able to manipulate the elements of an array of integers. The program acquires the array's content and then invokes two functions:

a. `avgVect`: computes the average value of the elements of the array and returns this value;
b. `upperLimit`: counts the number of elements which is greater than a certain limit and returns the count value.

The caller function has to print the average and the count values.

Hint: the prototype of the avgVect function will be:

```
float avgVect(int v[], int n);
```

the same way, the prototype of the upperLimit will be:

```
int upperLimit(int v[], int n, float limit);
```

Further insight: merge the two functions in order to obtain a single function that returns the average value and stores and the count of the elements greater than the average value into the parameter named `greater`, as defined in the following prototype:

```
float over_Avg(int v[], int n, int *greater);
```

Exercise 5. Write a C program that:

a. defines two variables of type *char*;
b. acquires their values from the keyboard;
c. check if the characters are both alphabetic:
   i. if yes, check also if they are equal: if not equal, print them in the alphabetic order;
   ii. otherwise, print a message that indicates if at least one of them is a digit.

Exercise 6. Write a C program that reads text from keyboard and prints it on the screen, by changing the first character of each word into uppercase.

For example, if the text provided as input is the following:

```
fatti non foste
per viver come bruti
ma per seguir virtute e canoscenza
```

The program has to appear on the screen as follows:

```
Fatti Non Foste
Per Viver Come Bruti
Ma Per Seguir Virtute E Canoscenza
```

Hint: use the method of the flags to determine whether you are inside or outside the word.

Exercise 7. [1]Write a C program that is able to analyze a text document composed of an unknown amount of lines. The document is inserted from keyboard by the user. The program has to show the following statistics:

a. the total number of lines;
b. the total number of words;
c. the average length of a word.

For the purposes of this problem, a "word" is defined as a sequence of characters without spaces, all appearing in the same line.

For example, if the text provided as input is the following:

```
fatti non foste
per viver come bruti
ma per seguir virtute e canoscenza
```

the program has to print the following values:

```
Number of lines: 3
Number of words: 13
Average length of a word: 4.5
```

Hint: modify the previous program (if well done, it will need minor changes).

[1] This exercise will be solved using a multimedia format, and its solution will be provided in the course site during the following weeks.