Ministry of Education of Republic of Moldova

Technical University of Moldova

Faculty of Computers, Informatics and Microelectronics

Software Engineering Department

Computer Programming

Laboratory work #4

One-Dimensional Array Operations and Processing

Author:
George Răbuș
std. gr. FAF-200

Verified:
Alexandru Furdui

Chișinău 2023

Theory Background

Strings and characters are fundamental elements in computer programming, often used to represent and manipulate text-based data. While a single character can be seen as the smallest unit of text, a string is a sequence of characters, forming a textual structure. In C language does not exist data type string and therefore do not exist variables of string type but it is possible to process strings. A string in C language represents a set of characters stored in statically or dynamically allocated 1-D array of character type and terminated by null byte (null character) and such a set of characters is considered as one object - a string.

The Task

Describe your task, and enumerate the task/tasks you have implemented:

1. Implement a function that reads a string and checks if it's a palindrome (reads the same forwards and backwards).
2. Implement a function that reads a string and converts it to uppercase.
3. Implement a function that reads a string and counts the number of vowels (a, e, i, o, u) in it.
4. Implement a function that reads a sentence from the user and counts the number of punctuation marks (e.g., periods, commas, question marks) in the sentence.

Technical implementation

Pseudo code here of base task with explanation

```
function basetask(s: string, size: integer):
    // Initialize variables
    declare words as array of strings
    declare word as string
    declare wordCount as integer
    declare sortedString as string

    // Print a prompt and read input
    print "Unsorted␣string:␣"
    read input into s with maximum length of size
```

```
    // Tokenize the string into words
    wordCount = 0
    word = tokenize(s, "␣\n")
    while word is not null:
        // Allocate memory for the word and copy it
        words[wordCount] = allocate memory for a string of
           length strlen(word) + 1
        copy word into words[wordCount]
        increment wordCount
        word = tokenize(null, "␣\n")

    // Sort the words in descending order by their lengths
    sort words in descending order based on length

    // Concatenate the sorted words into a single string
    sortedString = ""
    for i from 0 to wordCount - 1:
        append words[i] to sortedString
        if i < wordCount - 1:
            append "␣" to sortedString
        free memory for words[i] // Free memory allocated for
            each word

    // Print the sorted words in a row
    print "Sorted␣string:␣" + sortedString
```

Results

```
Write a sentence for basetask, Unsorted string: This is a sample sentance
Sorted string: sentance sample This is a
write a word for easy tasks, or sentence: racecar

It's a palindrome
The word with uppercases: RACECAR


The number of vowels: 3
Write a sentence to count the number of punctuation marks in it: Hello, World! You egg?
The number of the punctuation marks: 4
```

Sample figure

Conclusion

Here you go conclusions....

Bibliography

1. chatopenai.com
2. stackoverflow.com
3. geekforgeeks.com