

**Ministry of Education and Science**

**Technical University of Moldova  
CIM Faculty**

# **R E P O R T**

**Laboratory work #4**  
Procedures.  
V-11

Done by FAF-151 student:  
Spatari Stanislav

Verified by senior lecturer  
Sudacevschi Viorica

Chişinău 2017

## Laboratory work #4

### V-11

1. Elaborate a program that performs the search of the characters that correspond to the student name and surname in the defined string of characters.
2. Use software interrupts to visualize the results (arrange characters in a column, string, diagonal, change background color, use messages in case if the certain character is not finding).

#### Program code:

```
.data
String1 db 'Spatari Stanislav$'
String2 db 'Sjgndjarv$'
String3 db 20 DUP(?)
String4 db 20 DUP(?)
error db 10,13,'Some char not found',10,13,'Missing characters->$'
its_ok db 10,13,'The whole name was found$'
```

```
.code
```

```
init proc
```

```
    mov ax, @data
    mov ds, ax
```

```
    lea di, String1 ; move in di the EA of string1
    lea si, String2 ; move in si the EA of string2
    lea bp, String3 ; move in bx the EA of string3
    lea bx, String4 ; move in bx the EA of string4
    ret
```

```
init endp
```

```
findNewStrings proc
```

```
    L1:
        cmp [di], '$' ; check if the end is not reached for string1
        je the_end
```

```
    L2:
        cmp [si], '$' ; check if the end is not reached for string2
        je not_found

        mov dl, [di] ; mov in bl the current character from string1
        cmp dl, [si] ; compare with the current character from string2
        jne continueString2
```

```
        mov al, [di]
        mov [bp], al ;copies found element to String3
```

```
        jmp continueString1
```

```
continueString2:
    inc si ; go to the next character of String2
    jmp L2 ; loop
```

```
not_found:
```

```
mov [bp], '*'
```

```
mov al, [di]
```

```
mov [bx], al ; writes char that is not found to String4
```

```
inc bx ; goes to next index in String 4
```

```
continueString1:
```

```
inc di ; go to the next character of string1
```

```
inc bp
```

```
lea si, String2 ;will permite search in String2 from begining
```

```
jmp L1 ; repeat the search
```

```
the_end:
```

```
mov al, '$'
```

```
mov String3[di], al ;copies $ to String3
```

```
mov [bx], al ;copies $ to String4
```

```
ret
```

```
findNewStrings endp
```

```
printString proc
```

```
lea si, String3
```

```
mov ah, 03h ;this block gets the cursor position
```

```
int 10h
```

```
begin3:
```

```
mov ah, 02h ;this block set the cursor position
```

```
int 10h
```

```
cmp [si], '$' ;test if not end of string
```

```
je end3
```

```
mov ah, 09h ;this block sets color
```

```
mov al, [si] ;sets the char
```

```
mov bh, 0 ;page number
```

```
mov bl, 14h ;setting color
```

```
mov cx, 1 ;nr of char to print
```

```
int 10h
```

```
inc si
```

```
inc dl ;this will move cursor right one char
```

```
jmp begin3
```

```
end3:
```

```
call printCRLF
```

```
call printCRLF
```

```
ret
```

```
printString endp
```

```
printInColumn proc
```

```
lea si, String3
```

```
mov ah, 03h ;this block gets the cursor position  
int 10h
```

```
begin1:
```

```
    mov ah, 02h ;this block set the cursor position  
    int 10h
```

```
    cmp [si], '$' ;test if not end of string  
    je end1
```

```
    mov ah, 09h ;this block sets color  
    mov al, [si] ;sets the char  
    mov bh, 0    ;page number  
    mov bl, 47h ;setting color  
    mov cx, 1    ;nr of char to print  
    int 10h
```

```
    inc si  
    inc dh
```

```
    jmp begin1
```

```
end1:
```

```
    call printCRLF  
    ret
```

```
printInColumn endp
```

```
printInDiagonal proc
```

```
    lea si, String3
```

```
    mov ah, 03h ;this block gets the cursor position  
    int 10h
```

```
begin2:
```

```
    mov ah, 02h ;this block set the cursor position  
    int 10h
```

```
    cmp [si], '$' ;test if not end of string  
    je end2
```

```
    mov ah, 09h ;this block sets color  
    mov al, [si] ;sets the char  
    mov bh, 0    ;page number  
    mov bl, 39h ;setting color  
    mov cx, 1    ;nr of char to print  
    int 10h
```

```
    inc si  
    inc dh  
    inc dl
```

```

    jmp begin2

end2:
    call printCRLF
    ret
printInDiagonal endp

printLastMessage proc
    lea si, String4
    mov di, 0

stringSizeInc:
    cmp [si], '$'
    je end4

    inc di
    inc si
    jmp stringSizeInc

end4:
    cmp di, 0
    jne notFound ;when di=0 then String 4 is empty and print its_ok
                    ;else print error

    mov ah, 09h
    mov dx, offset its_ok
    int 21h
    jmp finish

notFound:
    mov ah, 09h
    mov dx, offset error
    int 21h

    ;printing missing chars
    lea si, String4
    mov ah, 03h ;this block gets the cursor position
    int 10h
printNextChar:
    cmp [si], '$'
    je finish

    mov ah, 02h ;this block set the cursor position
    int 10h
    mov ah, 09h ;this block sets color
    mov al, [si] ;sets the char
    mov bh, 0 ;page number
    mov bl, 47h ;setting color
    mov cx, 1 ;nr of char to print
    int 10h
    inc si ;goes to next char
    inc dl ;will move cursor to right
    jmp printNextChar

```

finish:

```
    ret
printLastMessage endp
```

```
printNotFoundChars proc
    mov dx, offset String4 ;print the string3
    mov ah, 09
    int 21h
    call printCRLF
    call printCRLF
    ret
printNotFoundChars endp
```

```
printCRLF proc
    mov ah, 02h
    mov dl, 0Ah ;line feed
    int 21h
    mov dl, 0Dh; caridge return
    int 21h
    ret
printCRLF endp
```

```
printLF proc
    mov ah, 02h
    mov dl, 0Ah ;line feed
    int 21h
    ret
printLF endp
```

start:

```
    call init
```

```
    call findNewStrings ;now we have string String3 with char found
                        ;and * were char not found
                        ;and String4 with char not found
```

```
    call printString
```

```
    call printInColumn
```

```
    mov dh, 2    ;this block will change position of the cursor
    mov dl, 2
    mov bh, 0
    mov ah, 02h
    int 10h
```

```
call printInDiagonal  
  
call printLastMessage  
mov ah, 4Ch  
int 21h  
end start
```