**מעבדה 7:**

מגישים:

איילון בן סימון – 312162951

סער ויקטור – 312392822

**קוד תרגיל 1:**

;lab7.asm

.MODEL SMALL

.STACK 100h

.DATA

;define variables

itrSize DW ?

temp DB 0

n DW 0

perm DB 13,10,'The strings are permutated',13,10 ,'$'

notperm DB 13,10,'The strings are not permutated',13,10 ,'$'

str1 DB 'abcddcb$'

str2 DB 'ddccbbaa$'

flag DW 0

.CODE

.386

MOV AX,@DATA

MOV DS,AX

;BX/DI are pointers to strings

MOV BX,OFFSET str1

CALL bubbleSortStr ;calling to function bubbleSortStr

MOV BX,OFFSET str2

CALL bubbleSortStr ;calling to function bubbleSortStr

MOV BX,OFFSET str1

MOV DI,OFFSET str2

CALL StrCmp ;calling to function StrCmp

;check if the strings are permutated or not (1 for yes 0 for not)

CMP AX,1

JE permlabel

;prints notperm string

MOV DX,OFFSET notperm

MOV AH,9

INT 21H

JMP endlabel

;prints perm string

permlabel:

MOV DX,OFFSET perm

MOV AH,9

INT 21H

;end program

endlabel:

MOV AH,4Ch

INT 21h

;check the size of the strings

getSize PROC NEAR

MOV DI,BX

MOV AX,0

loopcount:

;check if the strings have ended

CMP BYTE PTR [DI],'$'

JE endloop

INC AX

INC DI

JMP loopcount

endloop:

RET

getSize ENDP

;sort 1 character in a string

bubblePass PROC NEAR

MOV DI,BX

MOV AX,0

MOV CX,0

;DX=flag

MOV DX,0

MOV SI,-1

DEC itrSize

loopswap:

INC SI

CMP SI,itrSize

JE endloop2

MOV AL,[DI]

CMP AL,[DI+1]

JLE incdi

MOV DX,1

MOV AL,[DI]

MOV CL,[DI+1]

MOV [DI],CL

MOV [DI+1],AL

incdi:

INC DI

JMP loopswap

endloop2:

RET

bubblePass ENDP

;this function perform bubble sort

bubbleSortStr PROC NEAR

CALL getSize ;calling the function getsize

MOV itrSize,AX

sortStr:

CALL bubblePass ;calling the function getsize bubblePass

CMP DX,0

JE endloop3

LOOP sortStr

endloop3:

RET

bubbleSortStr ENDP

;check if the strings are equal

StrCmp PROC NEAR

MOV AX,0

loopcmp:

MOV AL,[BX]

CMP [DI],AL

JNE endloop4

INC DI

INC BX

CMP BYTE PTR [BX],'$'

JE label1

CMP BYTE PTR [DI],'$'

JE label2

JMP loopcmp

label1:

CMP BYTE PTR [DI],'$'

JNE endloop4

MOV AX,1

JMP endloop4

label2:

CMP BYTE PTR [BX],'$'

JNE endloop4

MOV AX,1

endloop4:

RET

StrCmp ENDP

END

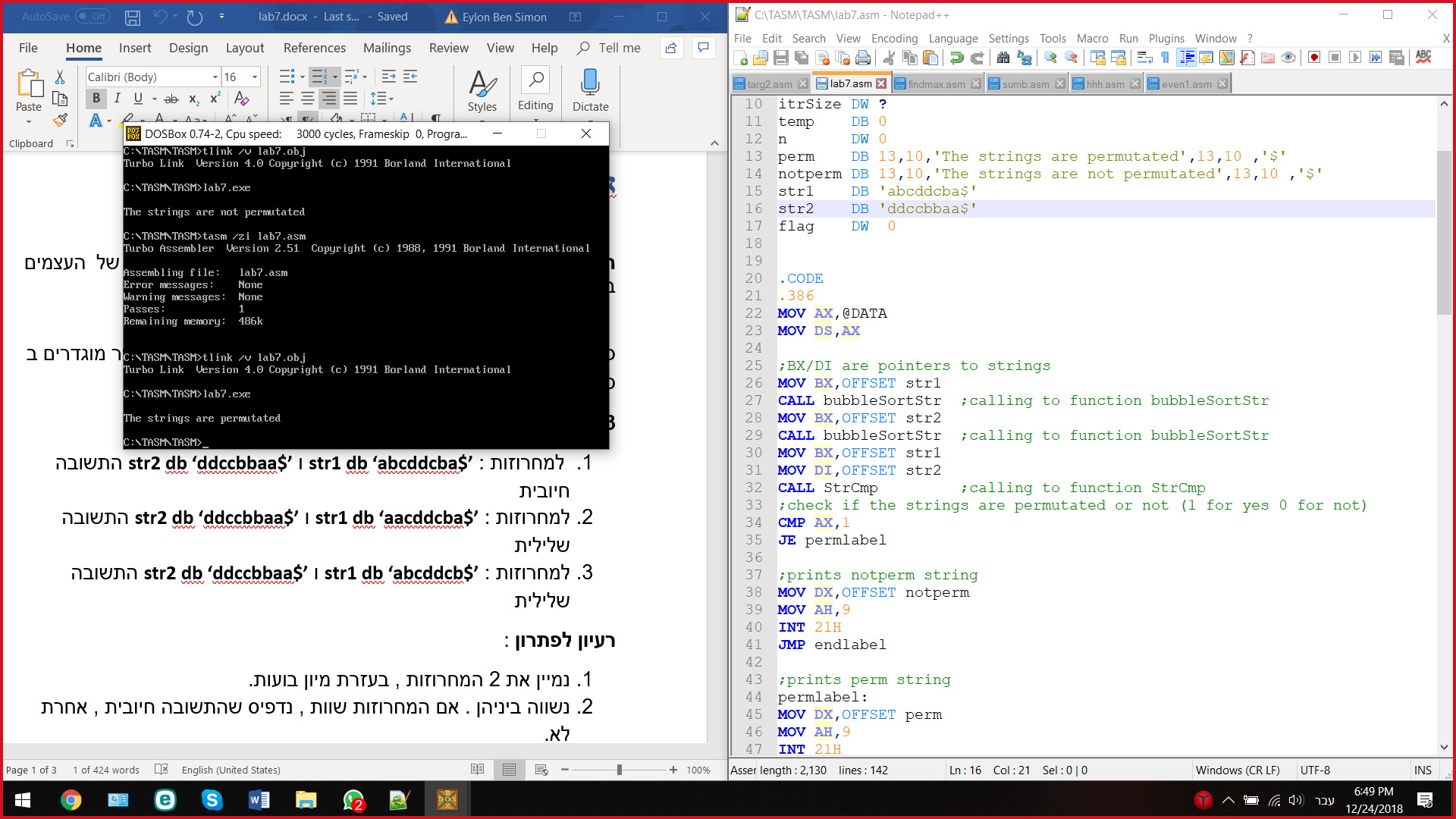
**פלט תרגיל 1:**

**3 דוגמאות לפלטים**  :

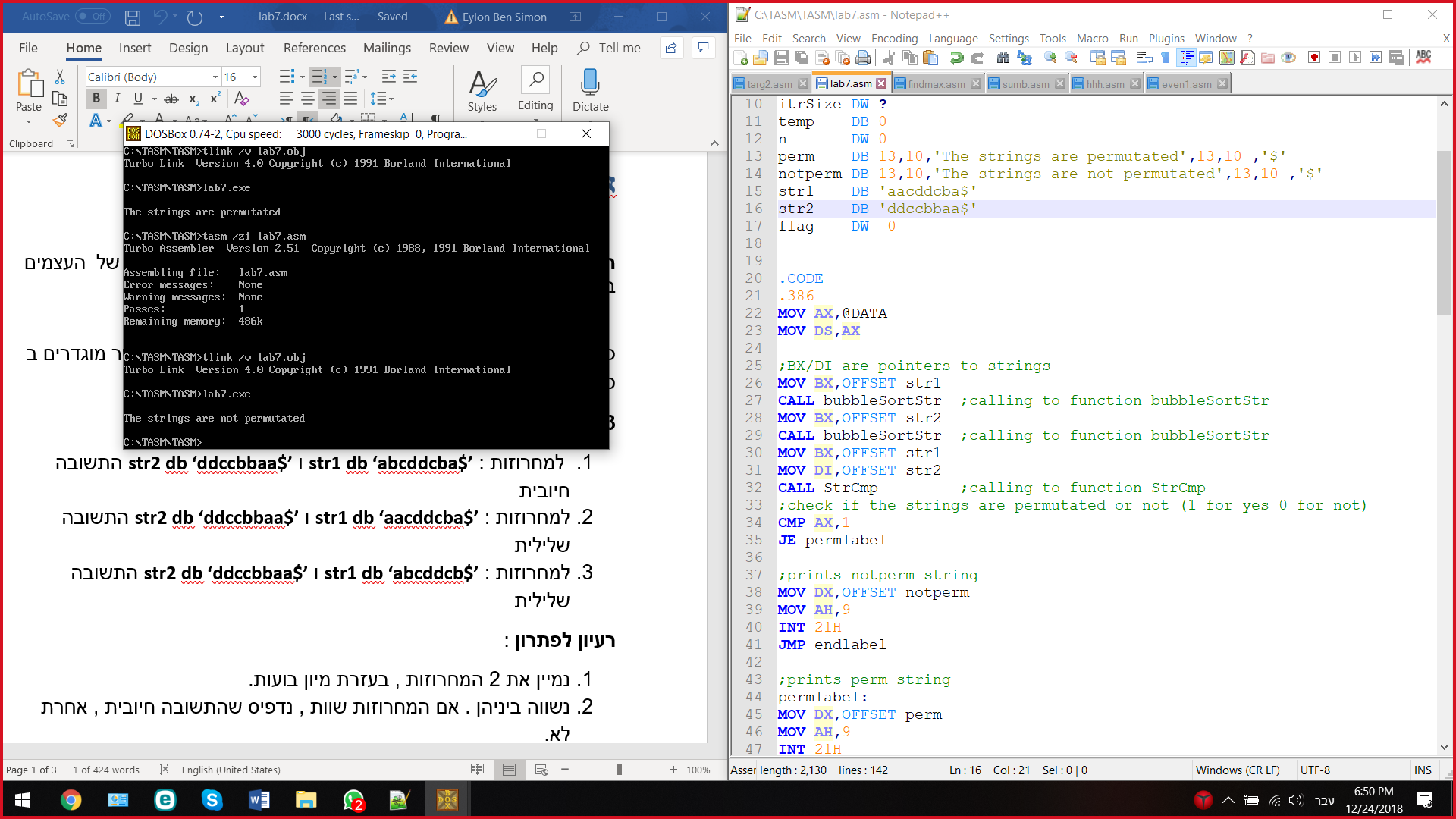
1. למחרוזות : **str1 db ‘abcddcba$’** ו **str2 db ‘ddccbbaa$’** התשובה חיובית
2. למחרוזות : **str1 db ‘aacddcba$’** ו **str2 db ‘ddccbbaa$’** התשובה שלילית
3. למחרוזות : **str1 db ‘abcddcb$’** ו **str2 db ‘ddccbbaa$’** התשובה שלילית

**דוגמא 1:**

מחרוזת 1:



**דוגמא 2:**



**דוגמא 3:**

