

תוכנית עיבוד מחרוזת והיפוך אותיות

```
; =====
; עיבוד 20 תווים, חישוב סכום ספרות וזיהוי תווים אחרים
; מטרה של השאלה: לקלוט 20 תווים, לסכום ערכי הספרות ולהמיר אותיות בגודלן
; וגם לספור תווים שאינם ספרות ולהציג את התוצאות
; קלט: 20 תווים והפלט: מחרוזת טקסט, סכום ספרות, מספר תווים אחרים
; =====

.model small
.stack 100h

.data

prompt      db 13,10,'Enter 20 characters:$'      ; הודעה להתחלת הקלט
nl           db 13,10,'$'                          ; מעבר שורה להדפסה
summsg       db 13,10,'numbers sum: ',0,0,0,'$'     ; מחרוזת להצגת סכום הספרות
othermsg     db 13,10,'other chars amount: ',0,0,0,'$' ; מחרוזת להצגת מספר תווים אחרים
textout      db 21 dup(?)                          ; מאגר לטקסט המומר
sumb         db 0                                  ; מצרף סכום הספרות מאותחל
othercnt     db 0                                  ; מונה תווים שאינם ספרות מאותחל

.code                               ; התחלת מקטע הקוד

main proc
    mov ax,@data                    ; טען את כתובת מקטע הנתונים
    mov ds,ax                      ; קבע את מקטע הנתונים בפועל

    mov ah,9                        ; פונקציה להדפסת מחרוזת
    lea dx,prompt                  ; קבע את מצביע ההודעה
    int 21h                        ; הדפס את ההודעה

    mov cx,20                       ; נאתחל מונה ל20 איטרציות
    lea di,textout                  ; קבע את מצביע המאגר לטקסט המומר

readlop:
    mov ah,1                        ; בחר פונקציה לקלט תו יחיד
    int 21h                         ; קלט תו והצג אותו ב AL

    cmp al,'0'                      ; השווה אם התו קטן מ0
    jb letterc                      ; אם כן, אז לא ספרה, נעבור לטיפול באותות
    cmp al,'9'                      ; השווה אם התו גדול מ9
    jbe digit                       ; אם בין '0' ל'9' אז התו הוא ספרה

letterc:
    cmp al,'A'                      ; 'A' השווה אם האות לפני
    jb spaceot                      ; אם כן ייתכן רווח או סימן מיוחד
    cmp al,'Z'                      ; 'Z' השווה אם האות עד
    jbe toggles                     ; בצע החלפת גודל, 'Z' ל' A' אם בין
    cmp al,'a'                      ; 'a' השווה אם האות לפני
    jb spaceot                      ; אם כן ייתכן רווח או סימן מיוחד
    cmp al,'z'                      ; 'z' השווה אם האות עד
    jbe toggles

spaceot:
```

```

    cmp al, ' '           ; השווה אם תו רווח
    je store              ; אם כן , אחסן כפי שהוא
    inc othercnt          ; אחרת , זה לא ספרה, נספור אותו
    jmp nextcha           ; נמשיך לעיבוד הבא

digit:
    sub al, '0'           ; (0-9) המרת תו ספרתי לערכו
    add sumb, al          ; הוספת הערך לסכום הספרות
    jmp nextcha

toggles:
    xor al, 20h           ; החלפת גודל אות

store:
    mov [di], al          ; אחסן את התו (ספרה או אות מותאמת)
    inc di                ; עדכן את מצביע המאגר

nextcha:
    loop readlop          ; חזור עד ש20 תווים עובדו

    mov byte ptr [di], '$' ; סימן סוף מחרוזת

    mov al, sumb           ; טען את סכום הספרות
    lea di, summsg+15      ; קבע את מיקום ההחלפה בהודעת הסכום
    call bytetodec3        ; המרת ערך ל3 ספרות באסקי

    mov al, othercnt       ; טען את מונה התווים האחרים
    lea di, othermsg+22    ; נקבע את מיקום ההחלפה בהודעת המונה
    call bytetodec3        ; ASCII המרת הערך ל-3 ספרות

    mov ah, 9              ; נבחר פונקציה להדפסת מחרוזת
    lea dx, nl             ; נקבע מצביע למעבר שורה
    int 21h               ; נדפיס רווח בין חלקי הפלט

    lea dx, textout        ; נקבע מצביע לטקסט המומר
    int 21h               ; נדפיס את הטקסט

    lea dx, summsg         ; נקבע מצביע להודעת הסכום
    int 21h               ; נדפיס את סכום הספרות

    lea dx, othermsg       ; נקבע מצביע להודעת המונה
    int 21h               ; נדפיס את מספר התווים האחרים

    mov ax, 4C00h          ; לבחר פונקציה ליציאה עם קוד חזרה 0
    int 21h               ; נחזור למערכת ההפעלה

main endp

; =====
; bytetodec3 פרוצדורות
; מטרה: המרת ערך 0-255 לשלוש ספרות
; =====

bytetodec3 proc
    push ax                ; שמירת באוגרים שהיו בשימוש

```

```

push bx

xor ah,ah          ; ל 0 AX אתחול חלק עליון של
mov bl,100         ; הגדרת בסיס לחלוקה במאה
div bl             ; שארית=AH, עשרות=AL מקבלים חילוק במאה

add al,'0'         ; המרת ספרה עשרות לאסקי
mov [di],al        ; אחסון ספרה ראשונה

mov al,ah          ; AL העברת השארית ל
xor ah,ah          ; ל 0 AH אתחול
mov bl,100         ; הגדרת בסיס לחלוקה לעשר
div bl             ; יחידות=AH, עשרות=AL מקבלים חילוק בעשר

add al,'0'         ; המרת ספרה שנייה לאסקי
mov [di+1],al      ; אחסון ספרה שנייה

add ah,'0'         ; המרת ספרה שלישית לאסקי
mov [di+2],ah      ; אחסון ספרה שלישית

cmp byte ptr [di],'0' ; אם ספרה עשרות היא 0
jne done          ; סיימנו
mov byte ptr [di],' ' ; נוסף רווח במקום 0
cmp byte ptr [di+1],'0' ; בדיקה לספרה השנייה
jne done
mov byte ptr [di+1],' ' ; הוספת רווח במקום 0

done:
pop bx            ; נשחרר אוגרים
pop ax
ret              ; חזרה לקריאה שממנה באנו

bytetodec3 endp   ; סוף פרוצדורה

END main          ; סיום קובץ התוכנית

```