

ארגון המחשב ושפת סף – תרגיל 2

סמסטר א', תשע"ט
צבי מלמד

מועד הגשה: : **עד ליום ראשון 30.12.18** בחצות (חצות שבין ראשון לשני).

הנחיות הגשה:

- **קבוצות עבודה:** יש לבצע את התרגיל לבד. אין הגשה משותפת!
- **אופן הגשה:** תיבת הגשה ב MAMA.
 - יש להגיש קבצי מקור בלבד. לכל שאלה בתרגיל יש לשלוח קובץ **asm**. **בלבד** של הפתרון וכן את קבצי ה data שעושים להם include מתוך התכנית. כלומר, **אין להגיש את תוצרי הקומפילציה** (למשל קבצי exe). במקום שמצוין יש להגיש גם את קובצי הנתונים
 - שמות הקבצים צריכים להיות: ex1_q1.asm, ex1_q2.asm, ex1_q3.asm.
 - את כל קבצי ההגשה יש לארוז בקובץ **ZIP**.
 - יש לרשום את השם ותעודת הזהות כהערה בקוד בראש כל קובץ ה- **asm**.
 - **יש לוודא שהתוכנית שאתם שולחים עוברת קומפילציה ומצליחה לרוץ – שאלה שלא עוברת קומפילציה, תקבל ציון 0.**
- **צילומי מסך:**
 - לכל שאלה עליכם לצרף צילום מסך של החלון שבו קימפלתם והרצתם את התרגיל. לשם כך השתמשו ב ALT+Print-screen או בכלי snipping tool של WINDOWS.
 - אין לשלוח צילום של כל המסך!
 - שמות הקבצים של צילומי המסך צריכים להיות כמו קבצי ה ASM רק עם סיומת png, למשל: ex1_q1.png, ex2_q2.png, ex1_q3.png
- **קבצי data:**
 - במקרה שנדרש להגיש בנפרד קובץ של מקטע הנתונים, שמו של הקובץ יהיה במתכונת ex1_q1_data.inc (למשל, עבור תרגיל #1 שאלה #1). יש לצרף קובץ זה לקובץ ה ZIP שאתם מגישים.
 - קובץ זה יכיל הגדרות נתונים בהתאם למוגדר בשאלה. התכנית צריכה לעבוד **נכון עם ערכים שונים של הנתונים**. כלומר, **בודק התרגיל, עשוי להשתמש בקובץ דומה, אבל עם ערכים שונים**, והתכנית (כמובן) צריכה לעבוד נכון.
 - במקרה שדרושים לכם נתונים נוספים, **שאינם מוגדרים בשאלה**, למשל, משתנים לתוצאות ביניים, **עליכם להגדיר קטע data נוסף, בתוך קובץ ה- asm**. שלכם. כמו כן, מחרוזת ההדפסה שכוללת את שמכם ותעודת הזהות, צריכה להיות בתוך ה asm. ולא בתוך קובץ ה data שעושים לו include.

קווים מנחים, לתשומת לבכם, המשפיעים על הניקוד:

1. יש לשים לב להוראות בנוגע להגדרת משתנים, העברת פרמטרים, התנהגות מבוקשת של התוכניות והפונקציות ומבנה הפלט שאמור להתקבל.
2. למען קריאות התרגיל יש לתעד בתחילת התרגיל מה התרגיל עושה. בתוכנית עצמה יש להקפיד על כתיבת הערות באנגלית.
3. כל הגשה באיחור מצריכה את אישור המרצה, ומראש.
4. איחור בהגשה ללא אישור: איחור בהגשה עד שבוע מפחית מהציון הסופי של התרגיל 10%. איחור מעבר לשבוע, מפחית 25%.
5. התוכנית צריכה לעבור קומפילציה ולהיות נכונה מבחינת syntax ולעבוד נכון!.
6. על התוכניות להישלח כ - Source Code משמעות הדבר הינה, שיש לשלוח את קבצי ה ASM בלבד ולא את שאר התוצרים. כפי שצוין לעיל, יש לצרף גם צילומי מסך של ההרצה.
7. בתרגילים שבהם מוגדרים קטעי נתונים – אין להסתפק בבדיקת נכונות עם הנתונים שמופיעים (מוגדרים) בקטע הנתונים הנתון. עליכם לבדוק את נכונות הפתרון שלכם עם נתונים נוספים, בהתאם לתנאי השאלה.

שאלה #1 סכום וקטורים (מערכים)

נתונים ארבעה וקטורים (מערכים), כולם בעלי אותו אורך (כלומר אותו מספר איברים).
 המערך Bvec הוא מערך של מספרים **לא-מסומנים** בגודל BYTE.
 המערך Wvec הוא מערך של מספרים **מסומנים** בגודל WORD.
 המערך Dvec הוא מערך של מספרים **מסומנים** בגודל DWORD.
 באזור הנתונים עליכם להכריז על מערך SumVec של DWORD שאיננו מאותחל, באורך זהה, והוא נועד להכיל את הסכום של שלושת הוקטורים, **כמספרים מסומנים**.

עליכם לכתוב תכנית שסוכמת את האיברים של שלושת הוקטורים לתוך sumVec.

כלומר:

```
SumVec[j] = Bvec[j] + Wvec[j] + Dvec[j];
```

שימו לב שבתהליך הזה, יש להמיר את הערכים בוקטור Bvec שאיננו מסומן, לערכים מסומנים. כלומר, אם קיים שם איבר בגודל בית, שערכו הוא 0FFh, אזי הוא יומר לערך +255.

לאחר שהתכנית סוכמת את כל הערכים לתוך הוקטור SumVec, היא מדפיסה הודעה מתאימה (ראו בדוגמא) ולאחר מכן, הדפסת המערך על ידי קריאה לפונקציה Print_Dword_Arr שמופיעה בהמשך.

התכנית שלכם תבצע include לקובץ בשם: **ex2_q1_data.inc**. קובץ זה יכיל את מקטע הנתונים הבא:

```
.data
Bvec    BYTE    0FFh, 0FEh, 10, 3
Wvec    SWORD   -5, -54, 100, 3000
Dvec    SDWORD  -500, 200, 300, 3000000
SumVec  SDWORD  lengthOf Bvec DUP (?)
msg1    BYTE    "Sum Vector is:", 10, 13, 0
```

עבור מקטע זה, הפלט של התכנית יהיה:

```
==> ex2_q1.exe
Tzvi Melamed id: 012345678 Ex2-Q1
Sum Vector is:
-250 +400 +410 +3003003
```

הנחיות:

- במידה ואתם צריכים הגדרות נוספות או מקטע נתונים נוסף (לדוגמא מחרוזת ההדפסה כל הכותרת כולל השם שלכם) עליכם להגדיר אותם בתוך קובץ התכנית שלכם (ולא בקובץ ה include).
- השתמשו בפונקציה writelnt להדפסת המספרים.
- השתמשו ב"מעט אוגרים".
- השתדלו לכתוב קוד קצר.
- בחרו שיטת (שיטות) מיעון אופטימלית לבעיה הנתונה.

- השתמשו בפונקציה הבאה (צרפו אותה לקוד) בכדי להדפיס את המערך. עליכם להציב את הערכים המתאימים לתוך האוגרים, ואז לכתוב קריאה לפונקציה ע"י הפקודה: `call Print_Dword_Arr`.

```
; Print_Dword_Arr: prints signed array of dwords
; Getting arguments in registers:
;   EDI - holds the address of the array (first element)
;   ECX - holds the number of elements
Print_Dword_Arr PROC
    push eax
    push ecx
    push edi
L1:
    mov eax, [edi]
    call WriteInt
    mov al, ' '
    call writeChar
    add edi, 4
    loop L1

    call CRLF
    pop edi
    pop ecx
    pop eax
    ret
Print_Dword_Arr ENDP
```

שאלה #2: החלפת תווים במחרוזת

נתונה מחרוזת בשם Seuss. עליכם לכתוב תכנית שקולטת מהקלט שלושה תווים. התכנית עוברת על המחרוזת, ובכל מקום שבו מופיע התו הראשון או התו השני שנקלטו, היא מחליפה את התו הזה בתו השלישי שנקלט.

לדוגמא אם המחרוזת היא abcdabcd והתווים שנקלטו מהמשתמש הם a ולאחר מכן d ולבסוף x אזי המחרוזת לאחר ההחלפה תהיה xbcxabcx.

להלן קטע קוד בשפת C++ שמבצע את המשימה הנדרשת:

```
char seuss[] = "some-value";
char first, second, third;
cin >> first >> second >> third;
for (int j = 0; j < ....; j++)
    if (seuss[j] == first || seuss[j] == second)
        seuss[j] = third;
```

בתחילה התכנית, לאחר שהיא כותבת לפלט את שמכם ות.ז. שלכם, התכנית כותבת את המחרוזת, ובסיום (לאחר ההחלפה) היא כותבת שוב את המחרוזת. עיינו בפלט ובמקטע ה-data למטה, בכדי להבין מהו הפלט שנדרש.

הנחיות:

א. את התווים עליכם לקלוט באמצעות קריאה לפונקציה ReadChar. להלן ההגדרה של הפונקציה: (קראו עליה עוד במידת הצורך בספר הקורס או באתר של Irvine: (Kip

ReadChar The ReadChar procedure reads a single character from the keyboard and returns the character in the AL register. The character is not echoed in the console window. Sample call:

```
.data
char BYTE ?
.code
call ReadChar
mov char,al
```

מכיוון שהפונקציה איננה כותבת לפלט את התו שהיא קלתה, כתבו אותו מפורשות ע"י קריאה לפונקציה writeChar. להלן דוגמת קטע קוד שקולטת את התו הראשון:

```
mov edx, OFFSET ask_1st
call writeString
call readChar
mov <destination>, al
call writeChar
call CRLF
```

ב. אין להשתמש בפקודת loop. עליכם לנהל את הלולאה באמצעות פקודות קפיצה או קפיצה מותנית.

ג. הקטע של הלולאה צריך להיות קצר ככל שניתן (במימוש שלי, הלולאה מכילה 8 פקודות).

התכנית שלכם תעבוד על מחרוזת במקטע הנתונים ששמה הוא Seuss.

המחרוזת מוגדרת בקובץ ex2_q2_data.inc. כלומר, התכנית שלכם תבצע include לקובץ בשם: ex2_q2_data.inc. (התכנית צריכה לעבוד גם עם מחרוזות שונה.

```
.data
Str_before BYTE "The string before is:", 10, 13, 0
Str_after  BYTE "The string after is:", 10, 13, 0
ask_1st    BYTE "Enter first char: ", 0
ask_2nd    BYTE "Enter second char: ", 0
ask_3rd    BYTE "Enter third char: ", 0
```

```
Seuss BYTE 10, 13,
    "'now! now!  hzve no fexr.", 10,13,
    "have no fexr!' szid the czt.", 10,13,
    "'my tricks zre not bxd,', 10,13,
    "szid the czt in the hxt.", 10,13, 10, 13, 0
```

להלן דוגמה של הרצת התכנית:

```
==>ex2_q2.exe
Tzvi Melamed id: 012345678 Ex2-Q2
The string before is:

'now! now!  hzve no fexr.
have no fexr!' szid the czt.
'my tricks zre not bxd,'
szid the czt in the hxt.

Enter first char: x
Enter second char: z
Enter third char: a
The string after is:

'now! now!  have no fear.
have no fear!' said the cat.
'my tricks are not bad,'
said the cat in the hat.
```

שאלה #3: מיון בועות

עליכם לכתוב תכנית שמבצעת מיון בועות על מערך מספרים מסומנים בגודל DWORD בשם Arr.

הקוד שלכם צריך להתאים לקוד הבא בשפת C:

```
#define LengthOf <גודל המערך>
for (j = 0; j < LengthOf - 1; j++)
    for (k = 0; k < LengthOf - j - 1; k++)
        if (Arr[k] > Arr[k+1])
            <swap>
```

בתחילת הריצה התכנית תכתוב הודעה מתאימה ואת המערך הבלתי ממוין.

לאחר שבוצע מיון המערך, התכנית תכתוב הודעה מתאימה ואת המערך הממוין.

בכדי לכתוב את המערך, השתמשו בפונקציה של שאלה #1 (העתיקו אותה לקובץ של השאלה הזאת)

המערך Arr מוגדר בקובץ ex2_q3_data.inc. כלומר, התכנית שלכם תבצע include לקובץ בשם: ex2_q3_data.inc. התכנית צריכה לעבוד גם עם מערך אחר באורך שונה, אבל באותו שם.

קובץ זה יכיל את מקטע הנתונים הבא:

```
.data
Str_before BYTE "The Array before is:", 10, 13, 0
Str_after  BYTE "The Array after is:", 10, 13, 0

Arr DWORD 10, 9, 8, 7, 1, 3, 5, 2, 4, 6
```

הנחיות:

- בשאלה זאת אין להשתמש בפקודה Loop. כלומר.. מן הסתם, כידוע, דרושה לולאה, ואפילו מקוננת בכדי לממש מיון בועות. **אבל בתרגיל הזה, עליכם לממש את הלולאות על ידי קפיצות מותנות וניהול עצמאי של המונים.**
- הפלט של התכנית צריך להיות לדוגמא כמו כאן:

```
==>ex2_q3.exe
Tzvi Melamed id: 012345678 Ex2-Q3
The Array before is:
+10 +9 +8 +7 +1 +3 +5 +2 +4 +6
The Array after is:
+1 +2 +3 +4 +5 +6 +7 +8 +9 +10
==>
```

בהצלחה!!