

ארכיטקטורת מחשב – תרגיל בית 03

אסמבלי – זיכרון, תנאים ולולאות

כללי:

נושאי התרגול:

- עבודה עם זיכרון
- כתיבת לוגיקה פשוטה באסמבלי

לאורך כל הסמסטר העבודה היא אישית - אסור לשבת לעבוד ביחד. עם זאת, מותר ואפילו מומלץ להתייעץ אחד עם השני. במידה ואתם מתקשים נסו להעזר בגוגל ובמידה ועדיין אתם מתקשים פנו למדריך!

הוראות להגשת התרגיל:

1. עבור כל התכניות שתכתבו באסמבלי, אנא השתמשו בתבנית הבאה:

```
org 100h  
[your code here]  
mov ah, 0  
int 16h  
ret
```

2. יש לשמור את הקבצים עם סיומת asm, כאשר שם הקובץ כולל את מספר השאלה. לדוגמא, תוכנית עבור שאלה מס' 1, ישמר בקובץ שנקרא 1.asm.
3. לאחר שכתבתם את הקוד, הריצו את תכניתכם ועקבו אחר פעולתה באמצעות כפתור emulate. וודאו כי תכניתכם פועלת כראוי ומבצעת את הנדרש.
4. יש להגיש קובץ zip (בלבד!) המכיל את כל התוכניות. יש להקפיד על השם של הקובץ לפי התבנית הבאה: **03_israel_israeli.zip** (כאשר 03 מציין את מס' העבודה, וישראל ישראלי הוא שם פרטי ומשפחה).

שאלה 1 - זיכרון

פתחו את הקובץ *addressing.asm* באמצעות emu8086. לפניכם שאלה עם חמישה חלקים המסומנים A,B,C,D,E. עקבו אחר ההוראות בכל חלק:

- א. חלק A – השתמשו בפעולות אריתמטיות על המשתנים num1,num2,num3 כדי להגיע לתוצאה הרצויה 23. אין להשתמש במספרים נוספים (immediates).
- ב. חלק B – הדפיסו את התו השלישי במערך התווים chrs. אל תשתמשו ישירות ב-chrs, אלא "הגיעו" למשתנה מתוך המצביע chrs_ptr.
- ג. חלק C – הדפיסו את הערך 15 ע"י שימוש באוגרים הנתונים ax=10, bx=4 ובפקודה lea (זהירות: חישוב הכתובות במעבד 8086 "רגיש" לאוגרים בהם משתמשים. כלומר אי אפשר לעשות למשל [bx+ax]). קרא באילו אוגרים מותר להשתמש).
- ד. חלק D – שנו את הגדרת המשתנים static_num1_ptr ו-static_num1_ptr שבראש התוכנית, כדי שהקוד הקיים של חלק D (ללא שינויים) ידפיס את הערך הנכון 7.
- ה. חלק E – עליכם לכתוב מספר שורות קוד ששימו ערכים נכונים למשתנים dynamic_num1_ptr ו-dynamic_num1_ptr כך שהקוד הקיים של חלק E ידפיס נכון את הערך 7. אסור לשנות את הגדרת המשתנים אלה ישירות, אלא רק לשנות את הערכים שלהם בזמן ריצה.

שאלה 2 – תנאי if else

כתבו תכנית אשר בודקת אם הערך שנמצא ב-ah גדול מהערך שנמצא ב-al. חיזרו על הבדיקה פעמיים: פעם אחת כאשר אתם מתייחסים לערכים כאל בלתי מסומנים (unsigned) ופעם אחרת כאשר אתם מתייחסים לערכים כאל מסומנים (signed). בשני המקרים (גדול \ קטן), התוכנית תדפיס הודעה מתאימה למשתמש. לדוגמה, עבור ah=0x7F, al=0x80:

Unsigned comparison: AH is smaller

Signed comparison: AH is bigger

שאלה 3 – if elif else

הוסיפו לשאלה 2 תמיכה במקרה שבו al ו-ah שווים.

שאלה 4 - while

קראו בספר על פעולות כפל וחילוק. כתבו תכנית אשר מחשבת (ומדפיסה) עצרת של מספר מסוים שערכו נמצא באוגר ax.

שאלה 5 - for

כתבו תכנית אשר מחשבת ומדפיסה את כמות הביטים הדולקים באוגר ax.
רמז: קראו מה עושה בדיוק הפעולה shl\shr וחשבו איך באמצעותה ובעזרת אחד הדגלים ניתן לפתור את השאלה.

שאלה 6 - nested for loop – לולאה מקוננת על מערך דו מימדי

כאשר מגדירים מערך דו מימדי באסמבלי, פשוט מקצים מערך חד מימדי שכמות האיברים שלו זהה לכמות האיברים של המערך הדו מימדי המבוקש. לדוגמה, נניח מערך דו מימדי של 3 שורות על 4 עמודות, כל איבר בגודל בית. נגדיר פשוט מערך בעל 12 איברים בגודל בית:

```
array db 12 dup(0)
```

בשפת אסמבלי אי אפשר לגשת למערך באמצעות מציין של שורה ועמודה, כמו לדוגמה array[N, M]. במקום זאת צריך לגשת לאיבר במיקום N כפול גודל השורה + M. לדוגמה, כדי לגשת לאיבר באינדקס 3 בשורה באינדקס 2, ניגש ל-array במיקום 2x4+3, כלומר array[11].

בתרגיל זה נחשב את עשרת השורות הראשונות במשולש פסקל https://en.wikipedia.org/wiki/Pascal%27s_triangle התחילו מהגדרת מערך בגודל 100 בתים (שיוכל לשמור 10 שורות כפול 10 עמודות). את הבית הראשון נאתחל להיות 1, כך:

```
array db 1, 99 dup(0)
```

כמו כן מומלץ להגדיר את המשתנים הבאים, על מנת לחסוך רגיסטרים ולהפוך את הקוד לקריא יותר:

```
row db 0
col db 0
size db 10
```

חשבו את 10 השורות הראשונות במשולש פסקל. לדוגמה 3 השורות הראשונות יהיו:

```
1 0 0 0 0 0 0 0 0 0
1 1 0 0 0 0 0 0 0 0
1 2 1 0 0 0 0 0 0 0
```

כדי להראות שהצלחתם, הדפיסו למסך את סכום כל האיברים שבשורה העשירית.

תזכורת- כדי להדפיס את ax ולאחריו רווח:

```
call PRINT_NUM_UN$
PRINT "
```

המשך בעמוד הבא

שאלה 7 – Switch (בונוס)

לפניכם קטע קוד שמשמש בפקודת switch :

```
switch (si)
{
case 1:
    printf( "Number One" );
    break;
case 2:
    printf( "Number Two" );
    break;
case 3:
    printf( "Number Three" );
    break;
case 4:
    printf( "Number Four" );
    break;
}
```

בשאלה זו תממשו קטע קוד זה באסמבלי ללא שימוש בתנאים (כלומר ללא שימוש בפקודות: ja, jl, jge ודומיהן), אלא ע"י שימוש בפקודת jmp ללא תנאי, ובמבנה מסוג branch table.

א. עיין בערך הויקיפדיה http://en.wikipedia.org/wiki/Branch_table

ב. ממש את קטע הקוד הנ"ל באסמבלי תוך שימוש בשיטה זו.

תזכורת: בשביל ביצוע הדפסות :

- i. שים את הקובץ magshimim.inc באותה התיקייה כמו הקוד שלך
 - ii. הוסף בסוף הקובץ את השורה include magshimim.inc
 - iii. בשביל להדפיס, השתמש בשורה PRINTN "Number One"
- הערה: אתה יכול להניח שאין צורך לטפל במקרה בו si מכיל מספר מחוץ לטווח 1-4.

המשך בעמוד הבא

רמז א

הרמז מוסתר מאחורי המלבן. הזז אותו כדי להציץ.

הזז אותי בשביל להציץ ברמז 😊

רמז ב

הרמז מוסתר מאחורי המלבן. הזז אותו כדי להציץ.

הזז אותי בשביל להציץ ברמז 😊

בהצלחה!