



ארגון ותכנות המחשב

234118

1

תרגיל מס' 1:

תא להחזרה:

תאריך הגשה:

מגייסים:

Yonibettan	302279138	וילם קון
דואר אלקטרוני	מספר ת.ז.	שם מלא

	203714902	זין ציון
דואר אלקטרוני	מספר ת.ז.	שם מלא

ציון

את"ם – תרגיל בית מס' 1 - סטודנט חורף תשע"ז

תאריך פרסום: 9.11.2016 תאריך הגשה: 23.11.2016 (23:55)

- ההגשה בזוגות בלבד לבד לתא ההגשה של הקורס ובאמצעות הגשה אלקטרוני.
- שאלות על התרגילים יש להפנות לחסן עבאסי hassan@cs.technion.ac.il.
- הגשות באיחור יש לתאמם עם חסן לפני מועד ההגשה הכללי.
- ערעורים והגשות באיחור יש להגיש אישיות לחסן. אין להגיש לתא הקורס לאחר מועד ההגשה.

בתרגילים זה נלמד להכיר את הסימולטור ע"י הקלדת תוכנית קצרירה והרצתה.
נתונה התוכנית:

`. = torg + 1000`

```
main: mov Ingth, r0      ;Initialize registers
      mov #arr1, r1
      mov #arr2, r2
      mov #sum, r3
      mov #diff, r4

loop:  mov (r1), (r3)    ;Read element from arr1
      add (r2), (r3)+   ;Read element from arr2 and compute result into sum
      mov (r1)+, (r4)   ;Read element from arr1
      sub (r2)+, (r4)+  ;Read element from arr2 and compute result into diff
      sub r0, loop      ;Go to next iteration
      halt

Ingth: .word 6
arr1:  .word 22, 11, 16., 14, 11, 200
arr2:  .word 2, 11, 20, 12, 7, 208
sum:   .blkw 6
diff:  .blkw 6

. = torg + 2000
halt
```

חלק א:

הוראות להרצאת התוכנית:

- הקיידו את התוכנית לקובץ טקסט רגיל וקרואו לו בשם `ex1.s11.ex`.
- השתמשו באסטמבלר ע"מ לתרגם את הקובץ הNIL לשפת מוכנה.
- הריצו את הסימולטור על מנת לבצע את התוכנית `ex1.s11.ex` שנוצרה ע"י האסטמבלר.

הוראות הגשה:

הגשה אלקטרוני באתר הקורס:

- **קובץ ex1.s11.ex בלבד!** יש להגיש את הקובץ עצמו (ולא מכווץ בתור קובץ Zip כפי שמצוין באתר).
-

הגשה יבשה לתא הקורס:

- הדפיסו את הקובץ `lst.ex1` שנוצר ע"י האסטמבלר (מעבר התוכנית המקורית).
- הדפיסו את פלט הסימולטור בזמן הרצת התוכנית, ע"י ייצרת קובץ `log.ex1` הכלול:
 - כל הפקודות שאתם מכנים (לייצר את הקובץ באופן אוטומטי).
 - תוכן האונגרים אחרי ביצוע שלב האיתחול ולפני ביצוע הלולאה.
- רמז: ניתן לקבוע `breakpoint` לפני התווית `loop`.
- תוכן המעריכים `sum` ו-`diff` לאחר הרצתה.
- תוכן האונגרים לאחר הרצתה.
- יש להדפיס את דף השער הנמצא באתר הקורס ולהדקו לדפים שאתם מגישים.

חלק ב:

להלן רישום של שינויים בתוכנית לעיל. עבור כל שינוי בדקו האם הוא גורם לשגיאה וכתבו האמ' :

- א. השגיאה מתגלחת בזמן התרגומים.
- ב. השגיאה מתגלחת בזמן הריצחה.
- ג. לא מתגלחת שגיאה אך התוכנית תנסה את אופן ריצתה ביחס לתוכנית המקורית.
- ד. לא יחול כל שינוי בריצת התוכנית.

יש להתייחס לכל שינוי בנפרד (ללא תלות בשינויים האחרים).

1. מוחקים את התוiotת "main:" (כלומר משאירים השורה **mov Ingth, r0** ללא התווiot) ומייצים את התוכנית בלחיצה על כפתור **continue** (או הקלדת החט **c** בשורת הפקודות)?

תשובה: ח' - (וונאי קאנאי) ועדי עכ' -

2. מוסיפים את השורה "r1 inc" לפני השורה שבה מופיעות התווiot "loop"?

תשובה: ק' - אונן און זלקי לאיאס ז' צ'ז'

3. מחליפים את השורה "main: mov (r1),(r3)" בתוכנה "loop: mov (r1),(r3)"?

תשובה: ק' - י' צ'ז' - האונאי צ'ז' גאנאלס'

4. מחליפים המספר "22" המופיע במערך arr1 ב- "18"?

תשובה: ק' - י' צ'ז' - האונאי צ'ז' גאנאלס' - 22

רמז: עבור כל סעיף מומלץ למחוק את קובץ ה- "lst". ואת קובץ ה- "11.o". לפני כל הריצה מחדש.

הדפיסו וצרפו את עמוד התשובות להגשה היבשה מחלק א.

בצלחה

Technion Computer Science Department -- PDP/11 X-Assembler Version (5.5) AA.

S O U R C E F I L E : ex1.s11

=====

000000		1	.= torg + 1000
		2	
001000 016700 000034		3	main: mov length, r0 ;initialize register
001004 012701 001042		4	mov #arr1, r1
001010 012702 001056		5	mov #arr2, r2
001014 012703 001072		6	mov #sum, r3
001020 012704 001106		7	mov #diff, r4
001024 005201		8	inc r1
		9	
001026 011113		10	loop: mov (r1), (r3) ;read element from arr1
001030 061223		11	add (r2), (r3)+ ;read element from arr2
and compute result into sum			
001032 012114		12	mov (r1)+, (r4) ;read element from arr1
001034 162224		13	sub (r2)+, (r4)+ ;read element from arr2
and compute result into diff			
001036 077005		14	sbb r0, loop ;go to next iteration
		15	
001040 000006		16	length: .word 6
001042 000022 000011 000020		17	arr1: .word 22, 11, 16., 14, 11, 200
000014 000011 000200			
001056 000002 000011 000020		18	arr2: .word 2, 11, 20, 12, 7, 208
000012 000007 000210			
001072 000000 000000 000000		19	sum: .blkw 6
001106 000000 000000 000000		20	diff: .blkw 6
		21	
001122		22	.= torg + 2000
002000 000000		23	halt

S Y M B O L T A B L E

=====

SYMBOL	VALUE	TYPE
-----	-----	-----
main	001000	text
length	001040	text
arr1	001042	text
arr2	001056	text
sum	001072	text
diff	001106	text
loop	001026	text

ex1

@ c

breakpoint at 001024 (001024)

@ r

r0 r1 r2 r3 r4 r5 sp pc priTNZVC IR time
000006 001040 001054 001070 001104 000000 001000 001024 0 00000 012704 11.

@ d001024

breakpoint deleted at 001024

@ c

HALT - HALT INSTRUCTION.

@ e sum, sum+12

addr	10	12	14	16	00	02	04	06
------	----	----	----	----	----	----	----	----

001070 000024 000022 000040 000026 000020 000410

@ e diff,diff+12

addr	04	06	10	12	14	16	00	02
------	----	----	----	----	----	----	----	----

001104 000020 000000 000000 000002 000002 177770

@ r

r0 r1 r2 r3 r4 r5 sp pc priTNZVC IR time
000000 001054 001070 001104 001120 000000 001004 016702 0 11010 000000 105.

@ L-