

## תרגיל בית מספר 2

**הגשה עד ל 16.05.2021 בחצות.** הגשה בבודדים בלבד. יש לכתוב פתרונות קצרים, מדויקים וברורים. שאלות 4 ו-6 תוגשנה בקובץ txt. שאר הפתרונות יוגשו בקובץ word או pdf. יש לבצע zip לשלושת הקבצים ולהגיש אותם יחדיו. קבצי ה-txt עבור שאלות 4 ו-6 יקראו בשמות: hw2\_q4.txt ו-hw2\_q6.txt בהתאמה.

התוכניות בתרגיל 4 ו-6 תיבדקנה על הסימולטור שנמצא במאמ. פתרונות שגויים (בשלב האסמבלר או הריצה), לא יתקבלו.

**שימו לב:** הפקודה SPA בסימולטור שונה מזו שנלמדה בהרצאה. בסימולטור הפקודה בודקת אם הערך ב AC גדול ממש מ-0 (ולא גדול או שווה ל-0 כפי שהוגדר בפרק 5). לכן בשאלות 4 ו-6 יש להשתמש ב SPA כפי שהוא מוגדר בסימולטור.

### שאלה 1 [12%]

צרו את כל הביטויים הבוליאניים שמאפסים ומקדמים את ה SC על פי הפקודות שלמדנו. שרטטו את המעגל הלוגי והראו מה ייכנס ל CLR של ה SC ומה ל INR שלו.

### שאלה 2 [13%]

נתונה התכנית הבאה, המתחילה מכתובת 100 (ORG 100) :

<b>ORG 100</b> BSA CNC HEX 500 HEX 700 BSA WLK HEX 500 STA BLA HLT BLA, HEX 0  <b>ORG 500</b> DEC 10 HEX 550 <b>ORG 550</b> DEC 20 HEX 600 <b>ORG 600</b> DEC 30 HEX 500 <b>ORG 700</b> DEC 40 HEX 750 <b>ORG 750</b> DEC 50 HEX 800 <b>ORG 800</b> DEC 60 HEX 700	<b>ORG 200</b> CNC, HEX 0 LDA CNC I STA HD1 ISZ CNC LDA CNC I STA HD2 ISZ CNC LDA HD1 STA PT1 LDA HD2 STA PT2 ISZ PT1 LDA PT1 I STA TM1 ISZ PT2 LDA PT2 I STA TM2 LDA TM1 STA PT2 I LDA TM2 STA PT1 I BUN CNC I  HD1, HEX 0 HD2, HEX 0 PT1, HEX 0 PT2, HEX 0 TM1, HEX 0 TM2, HEX 0	<b>ORG 300</b> WLK, HEX 0 LDA WLK I STA HDR STA PTR ISZ WLK  LOP, <u>(*)</u> ADD SUM STA SUM ISZ PTR LDA PTR I  <u>(*)</u> CMA INC ADD HDR  <u>(*)</u> BUN LOP LDA SUM BUN WLK I  SUM, DEC 0 HDR, HEX 0 PTR, HEX 0
---	---	---

- מה יהיה הערך של TM1 (בהקסה) בסוף ביצוע הסברוטניה CNC ?
- מה יהיה הערך של PT2 (בהקסה) בסוף ביצוע הסברוטניה CNC ?
- לגבי הכתובות החל מכתובת 500 (בהקסה) ומעלה, האם יהיו שינויים בסוף ביצוע הסברוטניה CNC ?
- מה מבצעת הסברוטניה CNC ?
- הסברוטניה WLK מסכמת אברי רשימה מעגלית. השלימו את הפקודות/ערכים הבאים:
  - בשורה (\*) צריכה להיות הפקודה
  - בשורה (\*) צריכה להיות הפקודה
  - בשורה (\*) צריכה להיות הפקודה
  - בסיום התכנית הערך של BLA בהקסה יהיה

שאלה 3 [12%]

נתונה התכנית הבאה ( התכנית מתחילה מכתובת 100 ). התכנית מקבלת תו המסומן ב CHR (בעמודה השמאלית) וכן מחרוזת המסומנת ב STR (בעמודה הימנית) ויושבת בכתובת 300 ומסתימת ב null (באפס).

שימו לב שהמחרוזת **מכווצת**, ז"א בכל מילה בת 16 סיביות ישנם שני תווים והסדר הוא קודם התו בבית העליון ואחר כך התו שבבית התחתון.

<b>ORG 100</b>	<b>ORG 200</b>	// cont. from column 2 **
LDA CHR	SRR, HEX 0	PTR, HEX 0
BSA SRR	STA C	C, HEX 0
HEX 0300	LDA SRR I	CNT, HEX 0
STA SL	STA PTR	MXC, DEC -1
HLT	ISZ SRR	MSK, HEX 00FF
SL, HEX 0	LOP, LDA LFT	LFT, HEX 0
<b>CHR,HEX 61</b>	SZA	
	BUN RGT	
	LDA PTR I	
	BSA SR8	
	BUN L2	<b>ORG 300</b>
	RGT, LDA PTR I	<b>STR,</b> HEX 7261
	AND MSK	HEX 7720
	L2, SZA	HEX 6372
	BUN L1	HEX 6162
	BUN ENS	HEX 206D
	L1, CMA	HEX 6561
	INC	HEX 6C00
	ADD C	
	SZA	
	BUN MOR	
	LDA CNT	
	STA MXC	
	MOR, ISZ CNT	
	LDA LFT	
	SZA	
	_____ // (*1*)	
	LDA LFT	
	CMA	
	STA LFT	
	BUN LOP	
	ENS, LDA C	
	SZA	
	BUN EN2	
	LDA CNT	
	STA MXC	
	EN2, LDA MXC	
	BUN SRR I // to be cont. **	
<b>ORG 400</b>		
SR8, HEX 0		
CIR		
CIR		
CIR		
CIR		
CIR		
CIR		
CIR		
CIR		
AND MS		
BUN SR8 I		
MS, HEX 00FF		

1. מה צריך להיות רשום בשורה המסומנת ב (\*1) ?
2. מה יהיה הערך ב SL בסוף התכנית ?
3. מה יהיה הערך של CNT עם סיום התכנית, עבור CHR שהושם בו 0 ?
4. מה יהיה הערך ב SL בסיום התכנית, עבור CHR שלא נמצא במחרוזת?
5. מה יהיה הערך ב SL בסיום התכנית, עבור מחרוזת ריקה?
6. עבור המחרוזת הבאה "ARCHITECTURE" המשוכנת בתוך STR כרגיל (שני תווים במילה), ועבור CHR=0 (אפס), מה יוחזר בכתובת SL בסיום התכנית?
7. עבור המחרוזת הבאה "ARCHITECTURE" המשוכנת בתוך STR כרגיל (שני תווים במילה), ועבור CHR='E', מה יוחזר בכתובת SL בסיום התכנית?

#### שאלה 4 [25%]

כתבו תכנית אשר מחסרת שני מספרים ארוכים A - B בני n מילים כל אחד, ( n מספר טבעי, כאשר  $1 \leq n \leq 64$  ) ומציבה את התוצאה C. המספרים יישמרו בזיכרון אופן הבא: נסמן שורה בזיכרון כמילה. המילה הנמוכה ביותר תהיה בכתובת הנמוכה ביותר, והמילה הגבוהה ביותר תהיה בכתובת הגבוהה ביותר, כאשר הכתובות עוקבות. התגיות AP, BP ו-CP יצינו בתכנית את הכתובות בהם מתחיל כל מספר.

דוגמא לייצוג המספרים (עבור n=4):

N, DEC 4  
AP, HEX 500  
BP, HEX 540  
CP, HEX 580

ORG 500      / A = DEF0 9ABC 5678 1234  
    HEX 1234  
    HEX 5678  
    HEX 9ABC  
    HEX DEF0

ORG 540      / B = 1234 5678 9ABC DEF0  
    HEX DEF0  
    HEX 9ABC  
    HEX 5678  
    HEX 1234

ORG 580  
    HEX 0  
    HEX 0  
    HEX 0  
    HEX 0

#### שאלה 5 [13%]

נתונה רוטינת שרות לפסיקות קלט/פלט אשר קוראת תווים מהקלט עד לקריאת התו "רווח", ואז מדפיסה את התווים (מלבד הרווח בסדר הפוך. בסיום ההדפסה הרוטינה תהיה מוכנה לקרוא שוב תווים עד לקריאת ה"רווח" ולהדפיס אותם לפי ההסבר לעיל. ניתן להניח כי:

- כל התווים שיוקשו מלבד ה"רווח" הם אותיות בלבד
- לכל הפחות יוקש תו אחד שאיננו רווח
- לכל היותר יקישו 20 תווים עד להקשת התו "רווח"
- אין שגיאות בקלט

קוד ה ASCII של הרווח הינו 32 דצימלי והוא נמצא כבר בכתובת המכונה MSP בהמשך.

לדוגמא, עבור רצף התווים dog ואחריו רווח, יודפס אחריו רצף התווים god . השלימו את השורות החסרות, בכדי שהדרישה שמעל תתבצע.

SRV,

```

ORG 100
STA SAC
CIL
STA SE
SKI
BUN LOP
INP
STA PTR I

```

BUN CNT  
ISZ PTR

LOP,

SKO  
BUN DON  
LDA CTR  
SZA  
BUN LP1  
BUN DON

```

LP1,   SNA
        BUN DON
        LDA PTR I
        OUT
        ISZ PTR
        ISZ CTR
        BUN DON
        BUN FIX

```

CNT, \_\_\_\_\_  
FIX, \_\_\_\_\_  
\_\_\_\_\_

DON LDA SE  
CIR  
LDA SAC  
ION  
BUN 0 I

ORG	200
SAC	DEC 0
SE	DEC 0
MSP	DEC -32
PTR	HEX 230
CTR	DEC 0
MON	DEC -1

כתבו תכנית אשר מקבלת שני מספרים שלמים N (מחולק) ו-D (מחלק) ומחזירה את תוצאת החילוק ב-Q ואת השארית ב-R. כך ש:  $Q \leftarrow \lfloor N / D \rfloor$  ו-  $R \leftarrow (N \bmod D)$  על פי האלגוריתם שיפורט בהמשך.

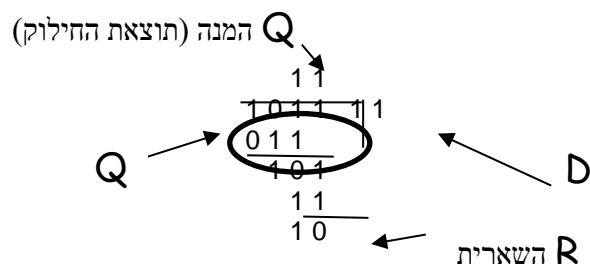
בניח ש  $N = 11$  (דצימלי) ו-  $D = 3$ . אחרי ביצוע התכנית, הערך של  $Q$  יהיה 3 ו-  $R$  יהיה 2

- שני המספרים המחולק N והמחלק D הם שלמים חיוביים.

- המחלק D לכלל היותר באורך של בית אחד (ז"א קטן ממש מ 256)

- המחולק  $N$  הוא מספר באורך 16 סיביות.

יש להשתמש בחילוק ארוך, לדוגמא:



האלגוריתם:

```
if D = 0 then error(DivisionByZeroException) end
Q := 0           -- Initialize quotient and remainder to zero
R := 0
for i := n - 1 .. 0 do -- Where n is number of bits in N
  R := R << 1      -- Left-shift R by 1 bit
  R(0) := N(i)     -- Set the least-significant bit of R equal to bit i of the numerator
  if R ≥ D then
    R := R - D
    Q(i) := 1
  end
end
end
```

בהצלחה!!