תרגיל בית מספר 2

הגשה עד 19/08/2021 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_q4.txt בהתאמה.

התוכניות בתרגיל 4 ו-6 תיבדקנה על הסימולטור שנמצא במאמא. 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, <u>המתחילה מכתובת 100</u> (ORG 100):

ORG 100 DRG 200 WLK, HEX 0 HEX 500 LDA CNC I HEX 700 STA HD1 STA HDR BSA WLK ISZ CNC LDA WLK ISTA HDR BSA WLK ISZ CNC ISZ CNC ISTA HDR STA PTR ISZ WLK LDA LDA WLK ISTA HDR STA HDR STA HDR STA PTR ISZ WLK LDP (*1*) ADD STA PTR ISZ WLK LDP MDD STA PTR ISZ PTR LDP LDP ADD STA STA PTR LDP LDP LDP MDD STA PTR INC INC STA												
HEX 500				CNC				 WIK				
HEX 700 STA HD1 ISZ CNC CNC ISZ WLK CNC ISZ CNC ISZ WLK CNC				OIVO,			1	V V L I X,				
BSA WLK HEX 500 LDA CNC I STA PTR ISZ WLK STA HD2 LDA HD1 STA PT1 LDA HD1 STA PT2 LDA PT2 LDA PT2 LDA PT2 LDA PT2 LDA PT2 LDA PT3 EVA PV4 EVA							I				•	
HEX 500 STA BLA BLA HLT ISZ CNC STA HD2 ISZ CNC CNC STA HD3 STA HD1 STA STA SUM STA STA SUM STA STA SUM STA												
STA BLA HLT												
HLT							ı		152	VVLK		(+4+)
BLA, HEX 0			BLA					LOP,	4 D D	CLIM		("1")
ORG 500 STA PT1 LDA HD2 STA PT2 LDA PTR I ISZ PTR LDA PTR I DEC 10 STA PT2 HEX 550 ISZ PT1 GMA INC ADD HDR INC ADD HDR INC ADD HDR ISZ PT2 STA TM1 ADD HDR ISZ PT2 STA TM2 LDA SUM BUN LOP LDA TM1 STA PT2 I SUM, DEC 0 DEC 30 STA TM1 BUN WLK I STA PT2 I SUM, DEC 0 DEC 40 LDA TM1 SUM, DEC 0 HDR, HEX 750 STA PT1 I SUM, DEC 0 HDR, HEX 0 STA PT1 I PTR, HEX 0 PTR, HEX 0 DEC 50 HD1, HEX 0 HD2, HEX 0 ORG 800 PT1, HEX 0 DEC 60 HD2, HEX 0 DEC 60 HTM, HEX 0 TM1, HEX 0 STA PT2 I SUM, DEC 0 HDR, HEX 0 DEC 60 HD2, HEX 0 TM1, HEX 0 TM1, HEX 0 TM1, HEX 0 DEC 60 TM1, HEX 0	DI 4		0									
ORG 500 LDA HD2 LDA PTR I DEC 10 STA PT2 CMA (*2*) HEX 550 ISZ PT1 INC INC ADD HDR INC ADD HDR	BLA,	HEX	0									
DEC 10												
HEX 550									LDA	PIR	ı	(+0+)
ORG 550 LDA PT1 I INC DEC 20 STA TM1 ADD HDR HEX 600 ISZ PT2 BUN LOP DEC 30 STA TM2 LDA SUM HEX 500 LDA TM1 BUN WLK I ORG 700 STA PT2 I SUM, DEC 0 DEC 0 DEC 40 LDA TM2 HDR, HEX 0 DTR, HEX 0 DTR, HEX 0 DTR, HEX 0 DTR, HEX 0 DTM, HEX 0											_	<u>(*2*)</u>
DEC 20 HEX 600 ORG 600 DEC 30 HEX 500 DEC 30 LDA PT2 I DRA TM1 ORG 700 DEC 40 HEX 750 ORG 750 DEC 50 HEX 800 ORG 800 DEC 50 HEX 800 DEC 60 HEX 700 TM1, HEX 0 HEX 700 STA TM2 BUN LOP LDA SUM BUN WLK I SUM, DEC 0 HDR, HEX 0 HDR, HEX 0 PTR, HEX 0 PTR, HEX 0 DEC 50 HD1, HEX 0 DEC 60 PT2, HEX 0 TM1, HEX 0							_					
HEX 600							I					
ORG 600 LDA PT2 I BUN LOP DEC 30 STA TM2 LDA SUM BUN WLK I HEX 500 LDA TM1 SUM, DEC 0 BUN WLK I I DEC 40 LDA TM2 HDR, HEX 0 HDR, HEX 0 PTR, HEX 0 PT1, HEX 0 PT2, HEX 0 PT2, HEX 0 TM1, M1, M2, M2, M2, M2, M2, M2, M2, M2, M2, M2									ADD	HDR		
DEC 30 HEX 500 DRG 700 DEC 40 HEX 750 ORG 750 DEC 50 HEX 800 ORG 800 DEC 60 DEC 60 HEX 700 STA TM2 BUN WLK I SUM, DEC 0 HDR, HEX 0 PTR, HEX 0 PTR, HEX 0 PTR, HEX 0 PTR, HEX 0 TM1, HEX 0 TM1, HEX 0 TM1, HEX 0												<u>(*3*)</u>
HEX 500 ORG 700 DEC 40 HEX 750 ORG 750 DEC 50 HEX 800 ORG 800 DEC 60 DEC 60 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 DEC 60 HEX 700 HEX 700 DEC 60 HEX 700 HEX 700 HEX 700 DEC 60 HEX 700 HEX 70							I					
ORG 700 STA PT2 I SUM, DEC 0 DEC 40 LDA TM2 HDR, HEX 0 HEX 750 BUN CNC I DEC 50 HD1, HEX 0 HEX 800 HD2, HEX 0 ORG 800 PT1, HEX 0 DEC 60 PT2, HEX 0 HEX 700 TM1, HEX 0												
DEC 40 HEX 750 ORG 750 DEC 50 HEX 800 ORG 800 DEC 60 DEC 60 HEX 700 DEC 60 HEX 700 LDA TM2 STA PT1 I BUN CNC I BUN											ı	
HEX 750 ORG 750 BUN CNC I DEC 50 HD1, HEX 0 HEX 800 HD2, HEX 0 ORG 800 PT1, HEX 0 DEC 60 PT2, HEX 0 HEX 700 TM1, HEX 0			700				I					
ORG 750 BUN CNC I DEC 50 HD1, HEX 0 HEX 800 HD2, HEX 0 ORG 800 PT1, HEX 0 DEC 60 PT2, HEX 0 HEX 700 TM1, HEX 0			-									
DEC 50								PTR,	HEX	0		
HEX 800			750			CNC	I					
ORG 800 PT1, HEX 0 DEC 60 PT2, HEX 0 HEX 700 TM1, HEX 0		DEC	50	HD1,	HEX	0						
DEC 60 PT2, HEX 0 HEX 700 TM1, HEX 0		HEX	800	HD2,	HEX	0						
HEX 700 TM1, HEX 0		ORG	800	PT1,	HEX	0						
		DEC	60	PT2,	HEX	0						
TM2 HEX 0		HEX	700	TM1,	HEX	0						
TIVIZ, TIEX				TM2,	HEX	0						

- 1. מה יהיה הערך של TM1 (בהקסה) בסוף ביצוע הסברוטינה CNC?
- 2. מה יהיה הערך של PT2 (בהקסה) בסוף ביצוע הסברוטינה 2CNC?
- 3. לגבי הכתובות החל מכתובת 500 (בהקסה) ומעלה, האם יהיו שינויים בסוף ביצוע הסברוטינה CNC?
 - ? CNC מה מבצעת הסברוטינה
 - 5. הסברוטינה WLK מסכמת אברי רשימה מעגלית. השלימו את הפקודות/ערכים הבאים:
 - a. בשורה (*1*) צריכה להיות הפקודה
 - b. בשורה (*2*) צריכה להיות הפקודה
 - בשורה (*3*) צריכה להיות הפקודה. c
 - d. 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.

	ORG 100	1		// conf	t. from column 2 **
	LDA CHR		ORG 200	PTR,	HEX 0
		SRR,	HEX 0		
	BSA SRR		STA C	С,	HEX 0
	HEX 0300		LDA SRR I	CNT,	HEX 0
	STA SL		STA PTR	MXC,	DEC -1
	HLT		ISZ SRR	MSK,	HEX 00FF
SL, HE	EX 0	LOP,	LDA LFT	LFT,	HEX 0
CHR,	HEX 61		SZA		
			BUN RGT		
			LDA PTR I		
			BSA SR8		ORG 300
			BUN L2	STR,	HEX 7261
		RGT,	LDA PTR I	3110,	HEX 7720
	000 400		AND MSK		
0.00	ORG 400	L2,	SZA		HEX 6372
SR8,	HEX 0		BUN L1		HEX 6162
	CIR		BUN ENS		HEX 206D
	CIR	L1,	CMA		HEX 6561
	CIR		INC		HEX 6C00
	CIR		ADD C		
	CIR		SZA		
	CIR		BUN MOR		
	CIR		LDA CNT		
	CIR		STA MXC		
	AND MS	MOR,	ISZ CNT		
	BUN SR8 I		LDA LFT		
Me			SZA		
MS,	HEX 00FF		// (*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. **		

- 1. מה צריך להיות רשום בשורה המסומנת ב (*1*)?
 - מה יהיה הערך ב SL בסוף התכנית?
- ? 0 שהושם בו CHR עם סיום התכנית , עבור CNT שהושם בו $^\circ$.
- 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?
- מה יוחזר , CHR='E' עבור המחרוזת הבאה "ARCHITECTURE" המשוכנת בתוך STR כרגיל (שני תווים במילה), ועבור "ARCHITECTURE" מה יוחזר בכתובת SL בסיום התכנית?

שאלה 4 [25%]

כתבו תכנית אשר מחסרת שני מספרים ארוכים A-B בני n מילים כל אחד, (מספר טבעי, כאשר $64 \le n \le A-B$ ומציבה את התוצאה C בכתובת

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 BP ,AP ו-CP יציינו 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
           / A = DEF0 9ABC 5678 1234
ORG 500
 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.

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

	ORG 100	LP1,	SNA	DON	LDA SE
SRV,	STA SAC		BUN DON		CIR
	CIL		LDA PTR I		LDA SAC
	STA SE		OUT		ION
	SKI		ISZ PTR		BUN O I
	BUN LOP		ISZ CTR		
	INP		BUN DON		
	STA PTR I		BUN FIX	ORG	200
	·	CNT,		SAC	DEC 0
		FIX,		SE	DEC 0
	BUN CNT		·	MSP	DEC -32
	ISZ PTR			PTR	HEX 230
	·			CTR	DEC 0
				MON	DEC -1
	·				
LOP,	SKO				
	BUN DON				
	LDA CTR				
	SZA				
	BUN LP1				
	BUN DON				

<u>שאלה 6 [25%]</u>

.R. מחַלֵק) וואת השארית ב Q אשר מקבלת שני מספרים שלמים און (מחוּלק) ווואר (מחַלֵק) מחזירה את תוצאת החילוק ב- Q ואת השארית ב $R \leftarrow (N \text{ mod } D)$ על פי האלגוריתם שיפורט בהמשך.

<u>דוגמא:</u>

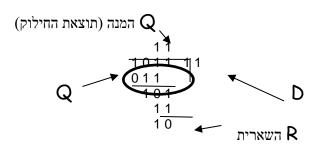
יהיה R-1 יהיה Q יהיה Q יהיה Q יהיה Q אחרי ביצוע התכנית, הערך של Q יהיה Q יהיה Q יהיה Q

<u>הנחות</u>:

- שני המספרים המחוּלק N והמחַלֵק D הם שלמים חיוביים.
- רכל היותר באורך של בית אחד (ז"א קטן ממש מ 256) D המחַלֵק D לכל היותר באורך של בית אחד
 - המחוּלק N הוא מספר באורך 16 סיביות.

<u>שיטת החילוק</u>

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 \geq D then 

R := R - D 

Q(i) := 1 end 

end
```

בהצלחה!!