

תרגיל בית 2 – הגשה ב-17/02/2024

הערות להגשה:

- את העבודה יש להגיש בזוגות בלבד.
- יש להגיש את התרגיל אך ורק מהמודל של **מאחד מבני הזוג** (אל תגישו שניכם בנפרד!).
- יש להגיש:
 - קובץ main.asm, תבנית לקוד מצורפת לתרגיל בקובץ נפרד.
 - קובץ asm עבור כל סברוטינה בנפרד:
 - בהמשך יוגדר עבור כל סברוטינה/קטע קוד, שתדרשו לכתוב, את הכתובת בזיכרון בה יתחיל המימוש.
 - קובץ submitters.txt עם הת"זים ואחוז ההשתתפות בהתאם לפירוט בהמשך.
 - במידה שהשתמשם בסברוטינות מתרגיל בית 1, יש לצרף את קבצי ה-asm שלהם.
- עליכם לציין בקובצי ה-asm וה-txt את תעודות הזהות ואת אחוז העבודה של כל שותף, באופן הבא בלבד:
 - ; 123456, 30%
 - ; 456789, 70%
- שימו לב לרווחים, פסיקים ונקודה פסיק שבתבנית לעיל. אל תחרגו ממנה!
- יתרה מזאת, בקבצי ה-asm, אין לבצע מרווח של שורה או יותר בין שורות הת"ז לבין שורת ה-orig.
- כפי שתראו בדוגמה המצורפת בעמוד האחרון של התרגיל.
- שימו לב, שותף אשר לא יהיה פעיל בתרגיל הבית עלול להיקנס בנקודות.**
- תהיה השגחה על כך ואף תיתכן בדיקה מדגמית של הבקיות שלכם בפתרון.**
- **ניתן להגיש באיחור של עד ארבעה ימים ללא עדכון צוות הקורס.** כל יום איחור יגרור 5 נקודות הורדה בציון.
- **לא תתקבלנה הגשות לאחר ארבעה ימים ממועד ההגשה.**
- אנא הקפידו על פתרון נקי, אלגנטי ומתועד לכל אורכו, אשר יסביר את הלוגיקה שלכם עם שמות לייבלים משמעותיים. שימו לב – זהו קוד אסמבלי, לכן תיעוד טוב אינו רשות, אלא **חובה!**
- על שמות הלייבלים של הסברוטינות/פונקציות שלכם להיות **בדיוק**, אחד לאחד, כפי שנכתב בתרגיל זה, יש לשים לב ל-case sensitivity של הלייבלים (באופן כללי).
- שימו לב, הטסטים נבדקים אחד אחרי השני ללא איפוס הסימולטור.
- **בדיקת תרגילי הבית הינה חצי אוטומטית. אי עמידה בדרישות הנ"ל עשויה לגרור הורדה בציון!**
 - אי עמידה בדרישות הפורמט אשר כולל:
 - שורות תעודת הזהות
 - שמות הקבצים
- תגרור הורדה של נקודות מסך הציון של תרגיל הבית.**
 - הדפסות של התרגיל צריכות להיות בהתאם לתבנית המוצגת בתרגיל.
 - **כל שורת פלט אשר אינה זהה לאחד לפלט המצופה תגרור הורדה של נקודה מהטסט.**
- שאלות בנוגע לתרגיל הבית יש להפנות לפורום.
- בעניינים אישיים ניתן לפנות למתרגל במייל:

AlonPapiniCSTA@gmail.com

שימו לב – סרטון הדרכה הוקלט לתרגיל בית זה כדי לעזור לכם, בקישור הבא:

<https://www.youtube.com/watch?v=72CL0SRQVrk>

- בתרגיל זה נתרגל עבודה עם מערכים, מחרוזות, קלט ופלט ב-LC3.
- (1) ממשו סברוטינה בשם **GetNum** - אשר תקלוט מספר חד/רב ספרתי ותדפיס למסך כל תו שנקלט. כלומר, הסברוטינה תקלוט את המספר תו אחר תו ע"י קריאה לפקודה *GETC*, כך שיבוצע לקלט *echo* למסך ע"י הפקודה *OUT*. הסברוטינה תחזיר ב-R2 את המספר הרב ספרתי שנקלט. קלט של המספר יסתיים בסוף שורה, כלומר ערך ה-ASCII 10 (לדוגמה, כשלחצתם על Enter במקלדת). הסברוטינה תתמוך במצבים הבאים:
- קליטת מספרים שליליים.
 - במידה שהוזנו תווים לא חוקיים, כלומר הקלט אינו מספר, יש להדפיס הודעה מתאימה ולקלוט מספר חדש.
 - יש לטפל במקרים של overflow מבחינת המספר שנקלט.
- מאחר שגודל הרגיסטר הינו 16 ביט, אזי במקרים בהם המספר שנקלט גדול מערכו המקסימלי האפשרי וערכו המינימאלי קטן מהערך השלילי האפשרי, יש להדפיס הודעה מתאימה ולקלוט מספר חדש. דוגמאות למספרים עם overflow:
- 35255
 - -215699
- דוגמאות למספרים בלי overflow:
- 8900
 - -278
- לא ניתן להניח מה יהיה מספר התווים שהמשתמש יקיש. יש לטפל במקרים בהם המשתמש מזין כמות גדולה מאוד של תווים.
- על הסברוטינה לתמוך בהרצות הבאות:

Enter an integer number: abc

Error! You did not enter a number. Please enter again: 8978455645239837849

Error! Number overflowed! Please enter again: -296

ממשו את הסברוטינה מכתובת X41F4.

הדרכה: הסברוטינה הזאת תהיה מורכבת יחסית לשאר התרגיל, ותהווה רוב העבודה שלכם. יש כאן הרבה עבודה סביב תווי ASCII, אז תצטרכו להשתמש בכמה ערכים ספציפיים של ASCII בשביל השוואות/חיבורים/חיסורים. אתם צריכים לקלוט תווים, תוך כדי התהליך לבדוק אם קלטתם איזשהו תו לא חוקי, או אם אתם קולטים מספר שיגיע ל-overflow. אם איכשהו הקלט שלכם שרד את כל זה, יש לכם את החלק שבו אתם בעצם מחשבים מספר סופי מכל הספרות שקיבלתם כקלט. תנשמו חזק ותחשבו אם אולי אתם רוצים לחלק את הפעולות השונות של הסברוטינה הזאת לתת-סברוטינות נפרדות. בנו לעצמכם פסאודו-קוד שנראה לכם סולידי לפני שאתם ניגשים למלאכה.

הערה: ניתן למעשה לממש את הסברוטינה הזאת **ללא** מערך עזר, או אפילו ללא שימוש בסברוטינת *Mul/Exp*! נסו לחשוב כיצד אתם יכולים לטפל בכל מקרי הקיצון שלכם מבלי לשמור את הקלטים שלכם למערך עזר. האם יש דברים שאפשר להתחשב בהם בכל רגע נתון?

- (2) ממשו סברוטינה בשם **PrintNum** - אשר מקבלת מספר רב ספרתי ב-R0 ומדפיסה אותו למסך. יש לתמוך גם במספרים שליליים.

ממשו את הסברוטינה מכתובת X4320.

הדרכה: סברוטינה זו אמורה להיות קצת קלה יותר בהשוואה לקודמת. כדי לפרק את המספר לספרות תרצו לחלק אותו ב-10 כל פעם עד שלא יישארו לכם ספרות (אז פה אפשר לקרוא ל-*Div* שלכם מתרגיל בית 1), ותוכלו להיעזר במערך עזר קטן פה. וודאו שאתם מפרקים את המספר לספרות תוך כדי התחשבות באפשרות שמדובר במספר שלילי, ושאתם מצליחים להדפיס את הספרות בסדר הצפוי (הספרה השמאלית ביותר קודם, או אפילו סימן המינוס לפניו אם המספר שלילי).

(3) ממשו סברוטינה בשם **Calculator** – המקבלת שני מספרים ברגיסטרים R0, R1. על הסברוטינה לשאול את המשתמש איזה פעולת חשבון ברצונו לבצע מבין הפעולות חיבור, חיסור, כפל, חילוק וחזקה. לאחר מכן, תדפיס למסך את התרגיל שנוצר, על פי התבניות שלהלן:
(שימו לב לדוגמת הפלט המצורפת בעמוד האחרון)

a+b

a-b

a*b

a/b (מציג אך ורק את המנה ללא שארית)

a^b

ניתן להניח כי תוצאת כל פעולה ניתנת לייצוג ב-16 ביט, וגם ניתן להניח שהקלט יהיה תקין.
כלומר, ביצוע כל פעולה על הערכים שיקלטו יניבו תוצאה בטווח [-32768, 32767].
יש להשתמש בסברוטינות מסעיפים 1,2 וכן בסברוטינות מתרגיל בית קודם.
לנוחיותכם, טבלת ערכי ה-ASCII נמצאת במודל.

ממשו את הסברוטינה מכתובת X4384.

(4) ממשו קוד המבצע קליטה של שני מספרים תקינים וחוקיים, לאחר מכן מבקש מהמשתמש לבחור פעולת חשבון עבורה ידפיס את התרגיל המתקבל מהקלטים. זהו קוד ה-main שלכם.
יש להשתמש בסברוטינות מסעיפים 1-3 וכן מתרגיל הבית הקודם.

ממשו את קוד ה-main מכתובת X3000.

דוגמה להרצה של הקוד:

```
Enter an integer number: 365498
Error! Number overflowed! Please enter again: 9
Enter an integer number: lod56
Error! You did not enter a number. Please enter again: oui777
Error! You did not enter a number. Please enter again: -- 27
Error! You did not enter a number. Please enter again: 2
Enter an arithmetic operation: /
9/2=4
```

מבנה הקוד main.asm המוגש יראה כך:

```
; ID of student 1, __%
; ID of student 2, __%
.ORIG x3000
Write here your 'main' code!
HALT
Write here your constant/labels/Auxiliary subroutine!
.END
```

מבנה הקוד של כל קובץ עבור סברוטינה שעליכם לממש יראה כך:

```
; ID of student 1, __%
; ID of student 2, __%
.ORIG X__
Name of the subroutine (example "Mul:")
Write here your code!
RET
Write here your constant/labels/Auxiliary subroutine!
.END
```