### הטכניון – מכון טכנולוגי לישראל

ארגון ותכנות המחשב

תרגיל 2 - חלק יבש

## המתרגל האחראי על התרגיל: איתי אילת.

שאלותיכם במייל בעניינים מנהלתיים בלבד, יופנו רק אליו.

כתבו בתיבת:**subject** יבש 2 את"ם.

## שאלות בעל-פה ייענו על ידי כל מתרגל.

הוראות הגשה:

### ● לכל שאלה יש לרשום את התשובה במקום המיועד לכך.

● יש לענות על גבי טופס התרגיל ולהגיש אותו באתר הקורס כקובץ .PDF

● על כל יום איחור או חלק ממנו, שאינו בתיאום עם המתרגל האחראי על התרגיל, יורדו 5

נקודות.

● הגשות באיחור יש לשלוח למייל של אחראי התרגיל בצירוף פרטים מלאים של המגישים

)שם+ת.ז.(

● שאלות הנוגעות לתרגיל יש לשאול דרך הפיאצה בלבד.

● ההגשה בזוגות.

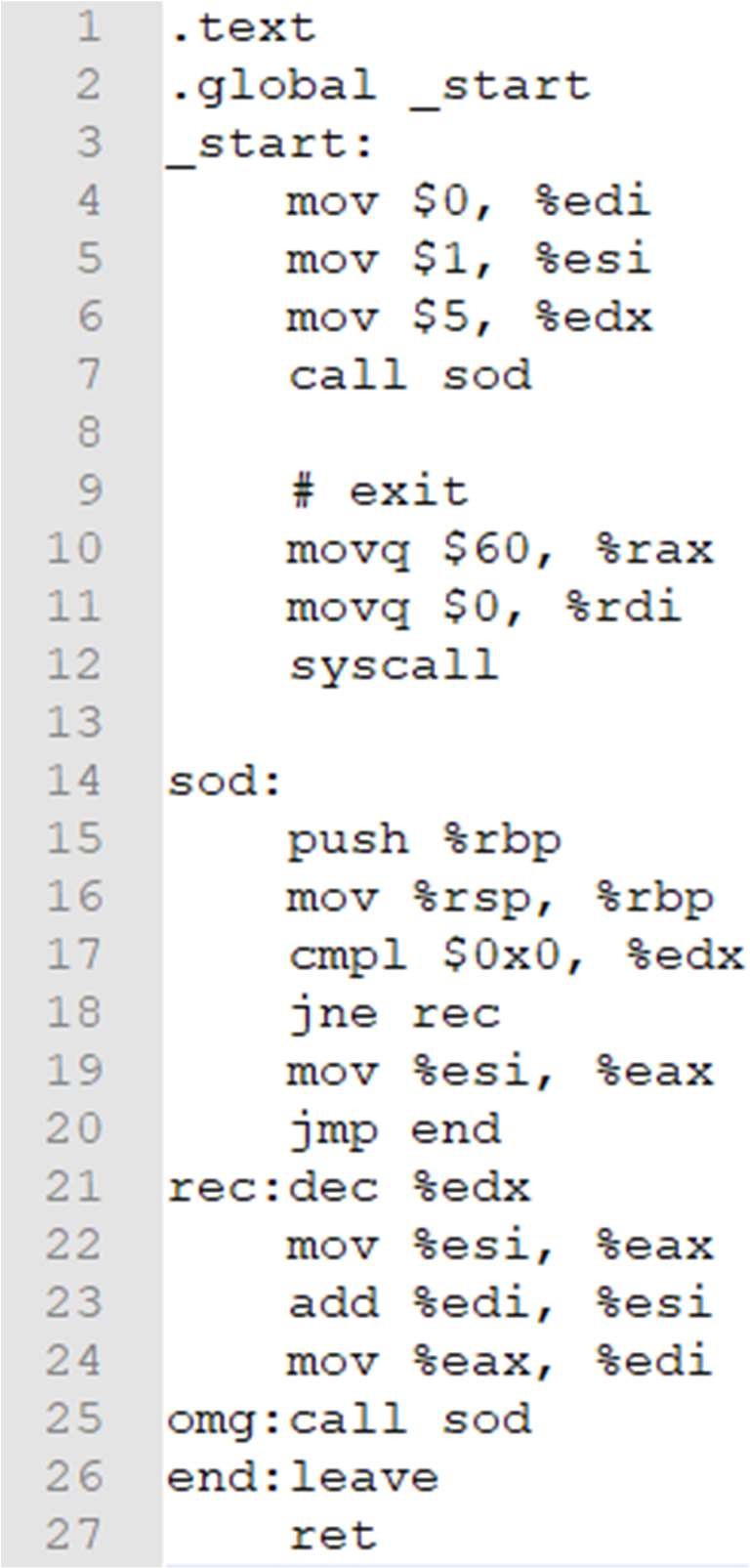
שגרות:

– ('נק

35)

# שאלה 1

:נתונה התוכנית הבאה בשפת אסמבלי



.1 מה פלט הפונקציה )בהתאם לנתונים שמופיעים בתוכנית.( 5) נקודות(

#### תשובה: 8

v Systemשנלמדה בתרגול? הקיפו בעיגול )או סמנו עם מרקר

.2 האם הפונקציה עומדת בקונבנציה

במחשב:( 3) נקודות(

כן / לא

.3 המירו את הפונקצייה לשפת c על ידי כך שתשלימו את המקומות החסרים בקוד 10) נקודות:( הערה: שלושת הפרמטרים צריכים להיות תואמים לשלושת הפרמטרים של פונקצית האסמבלי גם

מבחינת תפקיד וגם מבחינת סדר. כלומר, 1num צריך להתאים בתפקידו לפרמטר הראשון שמועבר

לפוקנציה בשפת אסמבלי גם מבחינת הקונבנצייה שלמדנו.

int sod ( int num1, int num2, int n){ If (n == 0)

return num2;

return sod( num2, num1+num2, n-1);

}

.4 הערה: אין צורך לדעת קידוד פקודות בשביל לפתור סעיף זה. אבל יש צורך להבין שפקודה שמורה

בזיכרון כרצף ביטים בדיוק כמו במקטע הנתונים ואחר כך המעבד מפרש את קידוד הפקודה לפקודה.

נתונה טבלה שמייצגת את קידוד הפקודות שנמצאות בשורות 22-24

|  |  |  |  |
| --- | --- | --- | --- |
| 24 | 23 | 22 | שורה |
| 0xc789 | 0xfe01 | 0xf089 | קידוד |

x4005eb0 והערך שבכתובות זו

החליטו להכניס בין השורה 19 לשורה 20 את הפקודות הבאות:

* 1. mov $omg-2, %rsi
  2. mov %rsi, 8(%rbp)

נתון שלאחר השינוי הכתובת של שורה 25 sod) call (omg: היא:

.8xe0 :הוא

.1 הקיפו בעיגול את התשובה הנכונה: 5) נקודות(

.a התוכנית כלל לא תיבנה. .b התוכנית תסיים את ריצתה אך פלט השגרה עלול להיות שונה מהפלט של התוכנית

המקורית

.c התוכנית לא תסיים את ריצתה לעולם.

.d התוכנית תסיים את ריצתה עקב שגיאה במהלך זמן ריצה.

.2 מה יהיה הערך של אוגר eax בסוף ריצת השגרה לאחר השינוי? שימו לב במידה ובסעיף הקודם רשמתם שהתוכנית לא מסתיימת עקב לולאה אינסופית רשמו מה יהיה הערך של האוגר אחרי

כמה איטרציות בלולאה. במידה ועניתם שהתוכנית תקרוס בזמן ריצה רשמו מה היה הערך של eax ברגע הקריסה. במידה ולדעתכם לא ניתן לדעת את הערך יש לרשום זאת! 5) נקודות(

#### תשובה: לאחר מספר איטרציות, 0x4005e9 = %eax )כתובת של שורה .(24

.5 )שימו לב השינוי בסעיף 4 לא תקף עבור סעיף זה( החליטו לשנות את הפקודה שבשורה 26

מleave לפקודה: %rsp $24, .add עבור אילו ערכי edx )הערך שedx מקבל בתוכנית הראשית. שורה (5 הפונקציה תחזיר את אותם

ערכים בדיוק כמו השגרה המקורית? הסבירו 7) נקודות(

#### תשובה: אבחנה1 – עומק הרקורסיה הוא rdx+1 אבחנה 2 – בראש המחסנית בהכרח מופיעים ebp או ra )אין במחסנית שימוש נוסף - local

**etc.. (variables בעת הנסיגה, אם נצליח לחזור ל \_start נקבל את אותו ערך החזרה כיוון שאין שינוי תוך כדי**

**ברגיסטר .%eax**

**חזרה כזו תתאפשר כמובן על גבי הebp השמורים במחסנית. עבור $24 ישנם שתי מופעים של ra ושל ebp )כל אחד-מופיעים לסירוגין( ולכן עבור מספר זוגי של רצפים כאלו נקבל ערך החזרה זהה. כלומר אם עומק הרקורסיה ) rdx+1 ( הינו זוגי נקבל**

**ערך החזרה זהה – אמ"מ rdx אי זוגי.**

קריאות מערכת:

– ('נק

38)

# שאלה 2

ג'ואי מרגיש מתוסכל מכך שחבריו חושבים שהוא פחות חכם מהם. לכן, הוא מחליט להרשים אותם בעזרת

כתיבת קוד אסמבלי.

.1 לפניכם מקטע הנתונים שג'ואי כתב:

section .data

msg1: .ascii "HOW YOOOU DOOIN?"

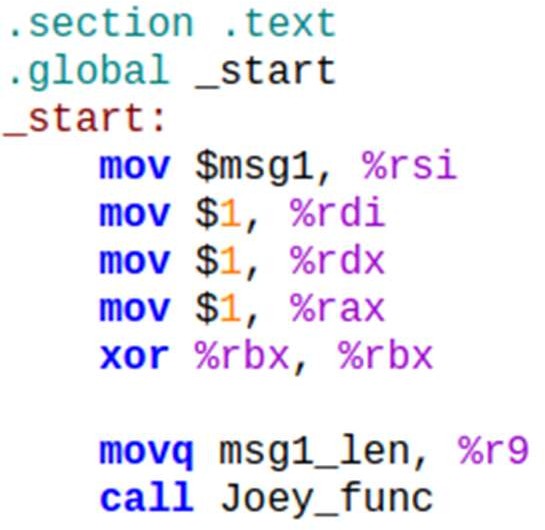
msg2: .ascii "JOEY DOESN'T SHARE FOOD!"

msg1\_len: .quad **msg2 - msg1** msg2\_len: .quad **msg1\_len - msg2** all\_msg\_len: .quad **msg1\_len - msg1**

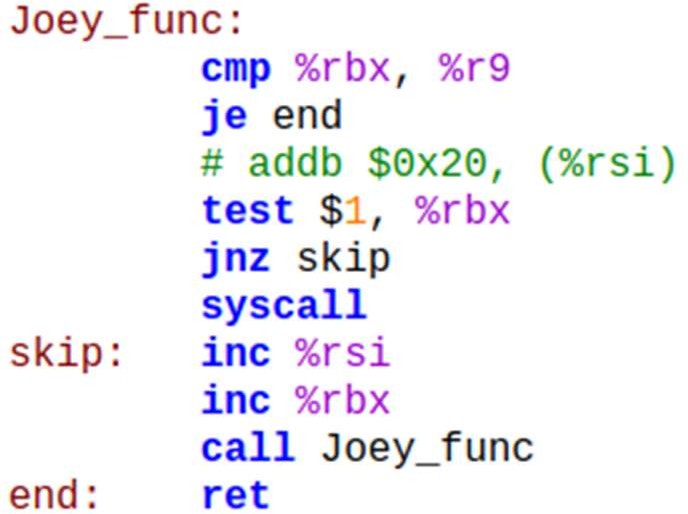
**ניתן לחלופין להגדיר משתנים אלו בעזרת הקבועים ,16 ,24 40 )מימין (msg1\_len**

עליכם להשלים את המקומות הריקים כך שמשתנה msg1\_len יהיה האורך של ,1msg בmsg2\_len יהיה האורך של 2msg ובמשתנה all\_msg\_len יהיה שווה לסכום אורכי המחרוזות

(נקודות 3) .2msgו 1msg

.2 כעת נתונה התוכנית הראשית שג'ואי כתב:

ובנוסף נתונה פוקנציה שהוא כתב:



מה יודפס בסיום ריצת הקוד? )שימו לב השורה השלישית בפונקציה נמצאת בהערה ולא רלוונטית

לסעיף.( 6) נקודות(

#### HWYOUDON :תשובה

.3 כעת מורידים את הסולמית שנמצאת בפונקציה )וכעת הפקודה חלק מהקוד( בנוסף מחליפים את

.9movq all\_msg\_len, %r :בשורה 9movq msg1\_len, %r השורה

הערה: שינויים אלו ילוו אותנו גם בסעיפים הבאים )בסעיפים **6 - 4** השינויים בסעיף **3** עדיין

תקפים.( מה יודפס כעת בסיום ריצת הקוד? 7) נקודות( **תשובה: hwyoudonje@osG@hr@od**

.4 בזמן שג'ואי אכל בסלון סנדוויץ, חיית המחמד שלו )אפרוח( טיילה על המקלדת והוסיפה את

הפקודה:

%r .9inc הפקודה נוספה שורה לפני הקריאה לפונקציה של ג'ואי בתוכנית הראשית.

מה יהיה פלט התוכנית כעת? 5) נקודות(

#### hwyoudonje@osG@hr@od0 :תשובה

.5 חברה טובה של ג'ואי פיבי אמרה לו ששימוש ברגיסטר 9r מביא מזל רע. ג'ואי נלחץ נורא והחליט שיש לבצע שינוי בקוד מבלי לשנות את תוצאות הפעולה של הפונקציה )כלומר הפלט צריך להיות

זהה.( כיוון ולא ידע איך לשנות את הקוד הוא החליט לבקש את עזרת חבריו. בסעיף הזה יופיעו העצות של כל החברים. עליכם לרשום ליד כל עצה האם היא לדעתכם תעזור

לג'ואי. נמקו בקצרה(!) 10) נקודות(

רייצ'ל מציעה להחליף את השימוש ב9r בשימוש ב.rdi

#### לא. rdi משמש כ descriptor ל syscall כלומר אם נשנה את ערכו מ,1 לא נקבל הדפסה

**למסך כפי שאנחנו מצפים.**

מוניקה מציעה להחליף את השימוש ב9r בשימוש ב.10r

#### כן. r10 משמש להעברת ארגומנטים לקריאות מערכת בדומה ל r9 ואין בו שימוש נוסף.

צ'נדלר מציע להחליף את השימוש ב9r בשימוש ב.11r

#### לא. בעת קריאה ל syscall הרגיסטר rflags מגובה לתוך r11 ולכן שימוש ברגיסטר זה יגרום

**למחיקת הגיבוי לדגלים וישנה את ריצת התוכנית.**

רוס מציע להחליף את השימוש ב9r בשימוש ב.rbp

#### כן. הצעה זו תעזור לנו לקבל פלט זהה אך אנחנו נפגע בקונבציה שבה אנו משתמשים מאחר ול rbp ישנו שימוש ייחודי – החזקת המסגרת הנוכחית במחסנית. מצד שני, בפונקציה של גו'אי אין כלל שימוש בפקודות שעושות מניפולציה על המחסנית כגון pop\push ולכן פלט הפונקציה לא ישתנה.

פיבי מציעה להחליף את השימוש ב9r בשימוש ב.rcx

#### לא. בעת קריאה ל syscall נבצע גיבוי לrip בתוך rcx ולכן אם נעביר את הערך שהיה ב r9 ל

**rcx ערך זה יידרס מיד בעת הקריאה .syscall**

.6 חבריו של ג'ואי מסבירים לו שהשימוש שלו ברקורסיה מיותר ובזבזני והוא יכול את אותו קוד בדיוק

לכתוב בלולאות. ג'ואי מחליט לבצע את השינויים הבאים:

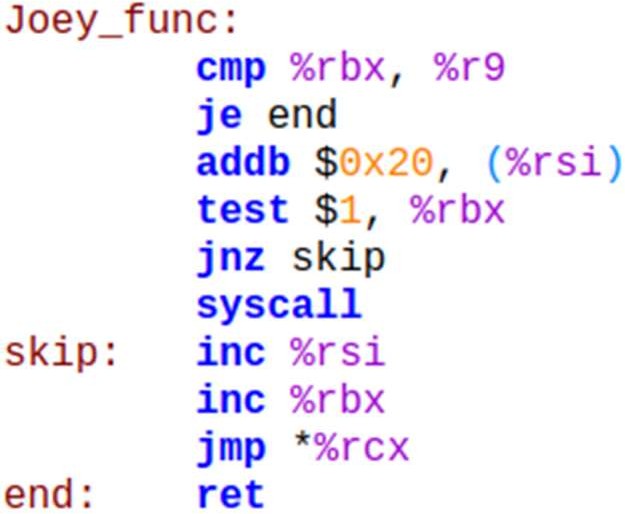
בתוכנית הראשית בשורה שלפני ביצוע הפקודה call ג'ואי מוסיף את הפקודה:

mov $Joey\_func, %rcx

ובתוך הפונקציה ג'ואי מוחק את השורה בה יש שימוש בפקודה call והחליף אותה בפקודה:

jmp \*%rcx

שימו לב שהתווית end נמצאת אחרי פקודה זו.

לצורך הבהרה הפונקציה נראית כך כעת:

כיצד שינוי זה משפיע על אופן ריצת הפונקציה. מה יודפס אם נריץ את הפונקציה? 7) נקודות(

#### תשובה: בעת הקריאה syscall מתבצע גיבוי לrip בתוך רגיסטר .rcx

**כלומר לאחר קריאת ה syscall הראשונה skip = rcx ותיווצר לולאה אינסופית.**

**באיטרציה הראשונה יודפס .h**

רמות הרשאה ואוגר הדגלים:

– ('נק

27)

# שאלה 3

.1 הפקודה pushfq דוחפת את הערך של אוגר הדגלים למחסנית. והפקודה popfq מוציאה את אוגר הדגלים מהמחסנית. הסבירו כיצד באמצעות שילוב של שתי פקודות אלו ניתן להדליק את

דגל .SF שימו לב במידה והדגל כבר דלוק יש להשאירו דלוק. אין לשנות את שאר הביטים

בריגסטר הדגלים. בנוסף, אין לשנות אף רגיסטר שהוא לא rsp rip, rflags, )גם לא באופן זמני.(

8) נקודות( הערה: במידה ובדקתם את עצמכם באמצעות דיבגר וראיתם שנדלק גם דגל TF זה בסדר

תלמדו בהמשך מדוע הוא נדלק תוך כדי דיבוג.

#### תשובה:

#### נשתמש ב pushfq על מנת לדחוף את אוגר הדגלים אל ראש המחסנית. לאחר מכן, נרצה לשנות רק את הביט המייצג את SF הרי הוא הביט השביעי.

#### על מנת לעשות זאת נבצע פעולת OR עם ראש המחסנית והמספר 128 הרי הוא 2 בחזקת 7-ובייצוג בינארי מספר זה יהיה מלא אפסים מלבד הביט השביעי. לאחר פעולה זו, אם הביט השביעי לא היה דלוק מלפני כן כעת הוא ידלק ואם הוא היה דלוק יישאר דלוק (מאופן פעולת OR). לבסוף, נוציא את ראש המחסנית בחזרה אל אוגר הדגלים באמצעות popfq.

**pushfq**

**orb $128, (%rsp) popfq**

.2 הולי התחמנית רוצה לאפשר לעצמה גישה ישירה אל התקני הקלט פלט ללא צורך בקריאות מערכת. איזה שינוי באוגר הדגלים יכול לעזור להולי במטרתה? 8) נקודות(

#### תשובה: שינוי IOPL לערך הדצימלי.3

.3 הולי מחליטה לנסות את התעלול מסעיף א' רק שבמקום לשנות את SF היא רוצה לשנות את .IOPL להפתעתה, היא לא מצליחה לשנות את הביטים הללו. הסבירו מה ההיגיון בכך שהיא לא

מצליחה לשנות את ?IOPL התייחסו לצורך בקריאות מערכת 5) נקודת( **תשובה: אילו כל משתמש היה יכול לשנות את IOPL אין כלל משמעות לרמות ההרשאה מאחר וכל משתמש יוכל להגדיר IOPL מקסימלי (3) ובזאת לאפשר לעצמו לבצע כל דבר**

#### שרק מערכת ההפעלה יכולה לבצע.

.4 הערה: סעיף זה קשור לפסיקות מומלץ לענות על סעיף זה לאחר התרגול על פסיקות.

וולי החבר המבולבל של הולי מתלבט כיצד ניתן לחסום פסיקות תוכנה לכן הוא שואל את הולי. אילו מבין התשובות הבאות על הולי לענות לו? יש לסמן את האפשרות הנכונה. 6) נקודות(

.1 כיבוי דגל IF באוגר הדגלים .2 הדלקת דגל IF באוגר הדגלים

00ל CPL שינוי .3

.4 לא ניתן לחסום פסיקות תוכנה.