

ארגון ותוכנות המחשב

תרגיל 2 - חלק י Bush

המתרגל האחראי על התרגיל: תומר צץ.

שאלותיכם במידע בעניינים מנהליים בלבד, יופנו רק אליו.

כתבו בתיבת subject: Bush 2 את"ם.

שאלות בעל-פה ייוננו על ידי כל מתרגל.

הוראות הגשה:

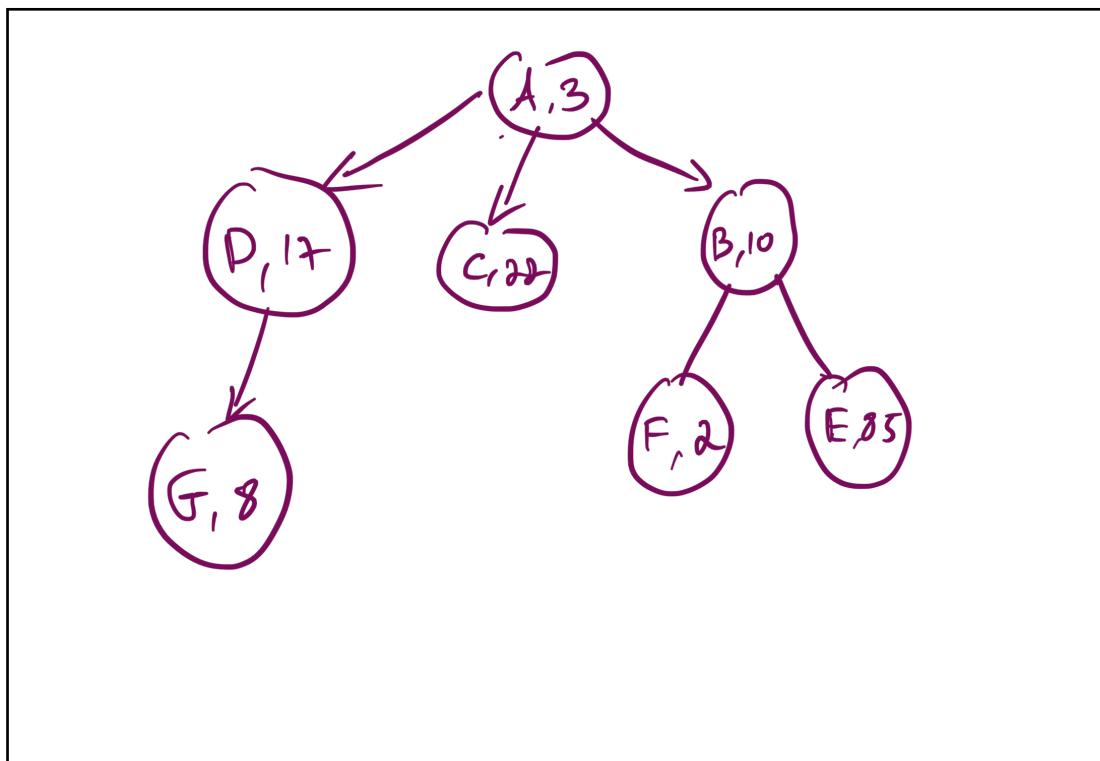
- לכל שאלה יש לרשום את התשובה במקום המועד לכך.
- יש לענות על גבי טופס התרגיל ולהגיש אותו באתר הקורס כקובץ PDF.
- על כל יום איחור או חלק ממנו, שאינו בתיום עם המתרגל האחראי על התרגיל, יורדו 5 נקודות.
- הגשתה באיחור יש לשלווה למיל של אחראי התרגיל ביצירוף פרטים מלאים של המציגים (שם+ת.ז.).
- שאלות הנוגעות לתרגיל יש לשאול דרך הפיאצה בלבד.
- ההגשה בזוגות.

שאלה 1 (45 נק') – שגרות:

נעה המ"מ שמעה כי היא הולכת לקבל מוחזר עתודאים. נעה לא יודעת מה עושים העתודאים בזמןם בחופשי אז היא החיליטה לרשום להם תוכנית באסמבלי לשעות הפנא. לפניכם מקטע הנתונים שנועה כתבה:

| | |
|----|----------------|
| 1 | .section .data |
| 2 | A: .long 3 |
| 3 | .quad B |
| 4 | .quad C |
| 5 | .quad D |
| 6 | .quad 0 |
| 7 | B: .long 10 |
| 8 | .quad E |
| 9 | .quad F |
| 10 | .quad 0 |
| 11 | C: .long 22 |
| 12 | .quad 0 |
| 13 | D: .long 17 |
| 14 | .quad G |
| 15 | .quad 0 |
| 16 | E: .long 85 |
| 17 | .quad 0 |
| 18 | F: .long 2 |
| 19 | .quad 0 |
| 20 | G: .long 8 |
| 21 | .quad 0 |
| 22 | |

- א. ציירו את הגרף המתkeletal מפירוש מקטע הנתונים (מומלץ להסתכל בתרגיל 3 תרגיל 1 ולהזכיר כיצד מפרשים את הדיזכרון כרשימה מקושרת). בכל צומת בגרף צינו את התווית המתאימה לו בלבד (אין צורך לציין ערכים נוספים) (3 נקודות)



הסמלת ספייר, שבאה לעזור לנויה, חשבה איך לשעשע את העתודאים הצעירים. לצורך כך, היא כתבה את הפונקציה הפשוטה `func` וקוד שמשתמש בא:

```

23  .section .text
24  .global _satart
25  _start:
26      mov $17, %esi
27      mov $A, %rdi
28      call func
29      movq $1, %rdi
30      sub %rax, %rdi
31      movq $60, %rax
32      syscall
33
34  func:
35      push %rbp
36      movq %rsp, %rbp
37      cmpl %esi, (%rdi)
38      jne next
39      mov $1, %rax
40      jmp end
41
42  next:
43      mov $0, %r10
44  test:
45      cmpq $0, 4(%rdi,%r10,8)
46      je fail
47      push %rdi
48      push %r10
49      mov 4(%rdi,%r10,8), %rdi
50      call func
51      pop %r10
52      pop %rdi
53      cmpq $0, %rax
54      jne finish
55      inc %r10
56      jmp test
57
58  finish:
59      mov $1, %rax
60      jmp end
61
62  fail:
63      mov $0, %rax
64  end:
65      leave
66      ret
67

```

ב. נתון שבתחילת התוכנית ערך של `rsp` הוא `X`. כאשר `X` הוא מספר הקסדצימלי. מה הוא הערך המקסימלי ומה הערך המינימלי ש-`rsp` יכול לארוך ריצת התוכנית? הבינו את התוצאה באמצעות נוסחה שהמספרים בה הם בסיס הקסדצימלי (בטאו את התשובה באמצעות `X`). (7 נקודות)

$$X - 56 \leq \text{rsp} \leq X$$



ג. רשמו מה יהיה פלט הפונקציה עבור קטע הקוד הנוכחי (5 נקודות) Main Cפּ - rdi - 0 (אייז טאנטן הום)

ד. המירו את הפונקציה לשפת C על ידי כך שתשלימו את המיקומות החסרים בקוד. היעדו בהגדרת `struct` שנთונה לכם (10 נקודות):

```

typedef struct _Node{
    int data;
    struct _Node** sons;
}Node;

```

הערות:

1. מבנה `struct` בזיכרון אליו היו היחסים מקרים את הקוד היה שונה מהצורה שבו הוא מופיע בקטע `data` לעיל. הסיבה טוביה לכך כמם בהערה הבאה.
2. הניחו שהכוונה ב`**Node` היא שרשימת המצביעים לבנים נמצאת בתוך `struct` ומוצגת כמערך באורך לא קבוע (ולכן גם `struct` בגודל לא קבוע). דבר זה אינו חוקי בשפת C. שפה C תופס מקום קבוע בזיכרון. לשם התרגיל אנחנו מיחים משוח חריג, שניתן להשיג רק בזכות העבודה שאנו כותבים באסמבלי)
3. מותר להשלים יותר ממילה אחת בכל קו אך לא יותר מפקודה אחת!

```

bool func ( Node *root, int x){
    if (root->data == X)
        return True;
    Node *son = root->sons;
    while(son != null) {
        if (func(son, x))
            return True;
        son += 1.
    }
    return False;
}

```

הערה: בסעיפים הבאים יש כל מיני שינויים בקוד. כל שינוי מתקיים רק בסעיף בו מופיע. זאת אומרת שהסעיפים לא תלויים אחד בשני.

ה. ציליל המ"כית מאמינה שהכל צריך להיות אחיד ומסודר. היא מחליטה לחת את מקטע הנתונים של נועה ולשנות בכל struct את הוגו longoa בquad. כמוום מקטע הנתונים ישתנה כך:

```

1 .section .data
2 A: .quad 3
3 .quad B
4 .quad C
5 .quad D
6 .quad 0
7 B: .quad 10
8 .quad E
9 .quad F
10 .quad 0

```

ובאופן דומה כל שאר האותיות יחליפו את הנתון הראשוני במקומם בquad. רשמו את השינויים שצריכים להיוות בקוד על מנת שייעבוד בצורה תקינה עם מקטע הנתונים החדש

(5 נקודות)

4(.rdi,%r198) 8(.rdi,%r198) -45,49
cmpd cmpq , 37 4(%r198)
esi fsi -2(%r198)

ו). יוגב הסמ"צ אוחב לעזר לנועה ולן החליט לשנות את מבנה הנתונים באופן הבא:

```
13    D: .long 17
14          .quad A
15          .quad 0
```

מה יהיה פלט התוכנית? יש לסמן תשובה מבין התשובות הבאות ולנמק במשפט אחד: (5 נקודות)

- a. התוכנית תשתים ופלט הפקנץיה יהיה
b. **התוכנית תשתים ופלט הפקנץיה יהיה**
c. התוכנית תכנס ללולאה אונסופית
d. התוכנית תקרוס במהלך ריצה
e. התוכנית כל לא תבנה

$$r_{di} = 1 - r_{ax} = 1 - 1 = 0$$

בימוק: גן אריה פולימר, רח' גראן, רח' גראן
הלאה (ב-ק'ם) א-ק'ם, א-ק'ם.

ז. הקוד הגיע למב"סית ובגלו קוצר הזמן שלא היה החלטה ל��ר את הקוד. לפניכם מספר שינויים שהמבחן סית הצעעה. עלייכם לכתוב עבור כל אחת מההצעות האם נכוןות השגורה תיפגע (האם יש קלט עבורי השגורה לאחר השינוי שוניה מהשגרה לפני השינוי). הסבירו בקצרה את תשובה! (10 נקודות)

- a. מחיקת הפקודת push וpop שבשורות 52 ו-47

לעומת הדרישה הנוכחית מ-152 מיליארדים דולר, קדרה מ-2010 מ-115 מיליארדים דולר, ו-100 מיליארדים דולר מ-2000.

- בשורה 51 מופיעות הפקודות `pop`.

הנתקן נספחים ל-3.5 מטרים. סדרת הנקודות מוצגת כסדרת נקודות נספחים ל-3.5 מטרים. סדרת הנקודות מוצגת כסדרת נקודות נספחים ל-3.5 מטרים.

- c. הוספה פקודה `push %rdi` אחרי `test` בשורה 44

הוספת פקודה ורץ push %ד בטרורה 44

- d. הוספה הפקודה `push %rdi` אחרי `test` בשורה 44, הוספה הפקודה: `mov %rsp, %rdi` לפני `jmp` בשורה 56.

הוספה הפקודה push אחרי test בשורה 44, הוספה הפקודה: mov (%rsp), %rdi לפני jmp בשורה 56.

שאלה 2 (30 נק') – קריאות מערכת:

ג'ואי מרגיש מתוכסכל מכך שחבריו חושבים שהוא פחות חכם מהם. לכן, הוא מחליט להרשים אותם בעצרת כתיבת קוד באסמבלי. הקוד שג'ואי מתכוון לכתוב קוד שיקרא קלט מהמשתמש שבו הוא יתאר אילו מאכלים הוא מעוניין לקבל והקוד יחזיר לו מה ג'ואי מוכן לתת לו.

- א. לפניים מקטע הנתונים שגואי כתוב מבלי ערכי הנתונים עצם:

```
.section .data
msg1: .ascii ???????
msg2: .ascii ???????
msg1_len: .quad _MSG2 - MSG1
msg2_len: .quad _MSG1_LEN - MSG2
all_msg_len: .quad _MSG1_LEN - MSG1
```

ג'ואי לא יודע עדין אילו מחרוזות הוא יכתוב. עליים להשלים את המקומות הריקים שקשורים לאורי המחרוזות כך שמשתנה `len` msg1 יהיה האורך של `msg1`, ב-`len2` msg2 יהיה האורך של `msg2` ובמשתנה `len_all` יהיה שווה לסכום אורכי המחרוזות `msg1` `msg2` ו-`msg3`. שימו לב עליים לעשות זאת בצורה כזו שהאורכים יהיו נכונים בעת ריצת התוכנית ללא קשר לאילו מחרוזות ג'ואי אתה כלערכיהם של `msg1` ו- `msg2`.

- ב. לפני ביצוע קריית מערכת מתבצעים גיבויים של ערכים מסוימים בתוכנית. נרצה להבדיל בין האחריות של מערכת הפעלה, אחריות המעבד ואחריות של קוד המשמש, בביצוע גיבויים אלו.

א. מה באחריות קוד המשתמש לגבות?

a. מה באחריות קוד המשמש לגבוט?

כפיגום x, rax, וען (אפיק) גם גבון מוגנה גם גבון גבון

b. מה באחריות מערכת הפעלה לגבוט?

c. מה באחריות המעבד לגבוט?

c. מה באחריות המעבד לגבי זהותם של ארגון כבש וארון כבש?

רשמו באיזה אחד משלבי התוכנית (קמפוס, ריצה, טעינה, קישור וכו') יקרה הכשלון ומדובר.

ד. חברה טובה של ג'ואי, פיבי, טוענת שאין בכלל צורך בשתי פקודות `return` שונות. ואמרה לג'ואי לבטל חלולותין את פקודת `sysretq` ולהשתמש רק ב-`ret`. האם ההצעה של פיבי טובה וישראל?

~~R11 → RELAGSI RCX → REP JC PCI
(2SF IC > NIC Wrapper → 2860)~~

רוא שמע על הקוד שג'ואי כותב לו ורוצה להציגו לחגיגה. הוא נסע לכנס מערכות הפעלה השנהית. וביקש אישור להוסיף קריית מערכת חדשה `pivot_sys`. קריית המערכת זו תיצור 3 תיקיות עם השם "pivot1", "pivot2", "pivot3" (הרשאות התיקייה לא משתנות). האם הקוד של רוא ש"י קרא על ידי `entry_syscall` כמו כל קריית מערכת צריך לשמור על קובננציות `V ?system` נמקו.

הסבירו מה על רוס לעשות/להוציא/לשנות לקוד מערכת הפעלה על מנת שהקראה שלו תעבוד (רק לפי מה שלמדתם בקורס)

גונטר, מנהל בית הקפה, שמע על כל המשחקים במערכת הפעלה שהחברים עושים וביקש מהם לכתוב עבורו גם קריית מערכת חדשה. הוא מעוניין בקריאת מערכת בשם `coffee_order`. היא תקבל בתוקן קלט את הארגומנטים הבאים: סוג קפה (מספר שלם), גודל הקוס (תו), ליקתן או לשבת (bool), חם או קר (bool), שם המזמין (מחרוזת), האם להוסיף מאפה (bool), האם התשלום באשריאנו במזומן (bool).

ומחזירה את מספר ההזמנה בתור פلت. האם ניתן להויסיף קרייאת מערכת שצוי? נמקו.

(Linux kernel / Linux Kernel Kernel Mod)

שאלה 3 (25 נק') – רמות הרשותה ואוגר הדגלים:

הפקודה `pushfq` דוחفت את הערך של אוגר הדגלים למחסנית. והפקודה `popf` מוציא את אוגר הדגלים מהמחסנית. הסבירו כיצד באמצעות שילוב של שתי פקודות אלו ניתן להציג את הדגלים `OF` ו-`CF`. שימושם במכשיר אחד הדגלים כבר דלוק יש להשאירו דלוק כלומר, בסיום התהליך על שני הדגלים להיות דלוקים. אין לשנות את שאר הביטים בריגיסטר הדגלים. בנוסף, אין לשנות אף רגיסטר שהוא לא `rflags`, `rip`, `rsp` (גם לא באופן זמני).

הערה: במידת ובדקתם את עצמכם באמצעות דיבאגר וראיתם שנדרך גם דגל TF זה בסדר תלמודו

במהשך מדוע הוא נדליך תוך כדי דיבוג.
 הפונקציית `main` מקבלת אינט (int) ומחזירה אינט (int).
 הפונקציית `sum` מקבלת אינט (int) ומחזירה אינט (int).

RFlags NC NW
AL DSZ. NZC גודל אוניברסיטאי
sf פ' -NSZ, DS RFlags

ב. הול' התחמנית רוצה לאפשר לעצמה גישה ישירה אל התקני הקולט פلت ללא צורך בקריאות מערכות. איזה שינוי באוגר הדגלים יכול לעזור להול' במטרתה?

הערה: לא צריך לציין פקודה ספציפית. רק להגיד מה צריך לעשות במסגרת התיאוריתית

ג. הול' מחליטה לנסוט את התעלול מסעיף א' רק שבמוקם לשנות את CF וOF היא רוצה לשנות את LOPL. להפתעתה, היא לא מצליחה לשנות את הביטים הללו. הסבירו מה ההיגיון בכך שהיא לא מצליחה לשנות את LOPL? התיחסו לצורך בקיאות מערכת.

לטראט של מילון עברי-נורווגי, שפורסם בשנת 1900 על ידי האקדמיה הנורווגית. מילון זה היה אחד ממקורותיו העיקריים של מילון זה.

ד. הול' לא מתייחסת ומנסה לgesht בלעה באופן אחר. היא תכתב עצמה קריית מערכת שתיעזר לה. הקרייה תאפשר לכל משתמש לקרוא לה ולאחר שהיא תסתהם המשתמש יוכל לgesht ישירות להתקני היקלט פטל ללא צורך בקריית מערכת. הסבירו מה על הול' לעשות בקריית

מערכת שהיא כתובת על מנת שהתוכנית שלה תצליח.

הפעלה��call היא פונקציית מילוק שפונה מ-
המערכת לתוכנה. הפעלה מושדרת ב-
המפלט הימני של ה-assembly. הפעלה
המוצפנת מושדרת ב-
המפלט השמאלי של ה-assembly. הפעלה
המוצפנת מושדרת ב-

הערה: הסעיפים הבאים קשורים לפסיקות מומלץ לענות עליהם לאחר תרגול 6.

ה. וויל החבר המבולבל של הווי מתלבט כיצד ניתן לחסום פסיקות תוכנה لكن הוא שואל את הווי.
אילו מבין התשובות הבאות על הווי לענות לו? יש לסמן את האפשרות הנכונה.

1. כיבוי דגל IF באוגר הדגלים
2. הדלקת דגל IF באוגר הדגלים
3. שינוי CPL ל00
4. לא ניתן לחסום פסיקות תוכנה.

ו. כתעת נתון שוויל הצליח להגיע למצב שבו CPL שווה ל0. וויל מעוניין לחסום פסיקות חומרה שאין מועברות דרך כניסה NMI. כיצד הוא יכול לעשות זאת? (5 נקודות)

1. כיבוי דגל IF באוגר הדגלים
2. הדלקת דגל IF באוגר הדגלים
3. עליו לחבר את הפסיקות לכנית NMI ואז לכבות את דגל IF
4. לא ניתן לחסום פסיקות חומרה ולכן לא יצליח.