ארגון ותכנות המחשב

תרגיל 3 - חלק רטוב

המתרגלת האחראית על התרגיל: גל קפצנל.

שאלות על התרגיל – ב- Piazza בלבד.

~~ התרגיל מורכב משני חלקים בלתי תלויים ~~

הוראות הגשה:

- ההגשה בזוגות.
- על כל יום איחור או חלק ממנו, שאינו באישור מראש, יורדו 5 נקודות.
    - ניתן לאחר ב-3 ימים לכל היותר.
- הגשות באיחור יתבצעו דרך אתר הקורס.
- הוראות הגשה נוספות מופיעות בסוף התרגיל. אנא קראו בעיון.

חלק א – Reading ELF Files

מבוא

חלק זה בתרגיל דורש היכרות עם תרגול 8. אתם תכתבו קוד בשפת C שיקרא קבצי ELF ויפענח אותם.

record scratch

freeze frame

רגע, מה? אבל למדנו על readelf שיודע לקרוא קבצי ELF. למה אנחנו צריכים לעשות זאת בעצמנו?

טוב ששאלתם! יש לזה מספר סיבות, אבל הנה שלוש הסיבות המרכזיות:

1. זה יעזור לכם להבין טוב יותר את מבנה קבצי ELF, חומר לא פשוט בכלל בפני עצמו ובטח כחומר עיוני בלבד.
2. זה יעזור לנו לדעת שהפנמתם את החומר הנ"ל טוב יותר.
3. ועוד סיבה טובה, והיא שתצטרכו את הפונקציה שתכתבו כאן. מתי? בתרגיל בית 4.

כתיבת הפונקציה

אתם תממשו בשפת C את הפונקציה **find_symbol** בקובץ **hw3_part1.c**. חתימת הפונקציה היא:

```
unsigned long find_symbol(char* symbol_name, char* exe_file_name, int* error_val);
```

כאשר:

- **exe_file_name** – הקובץ שאותו אנחנו קוראים, קובץ ה-ELF.
  <u>הערה:</u> אין צורך לבדוק שהוא ELF. אבל איך הייתם בודקים זאת אם הייתם צריכים?
- **symbol_name** – הסמל שאותו אנחנו מחפשים.
- **error_val** – מצביע (אפשר להניח שאינו NULL) שצריך למלא את הכתובת שאליה הוא מצביע בערך מסוים, על פי המפתח הבא:
  - **1** – אם הסמל נמצא, הוא גלובלי, ומוגדר בקובץ הריצה. כלומר, אם אין שגיאה.
  - **-1** – אם הסמל לא קיים בטבלת הסמלים בכלל.
  - **-2** – אם הסמל קיים, אבל רק כסמל לוקאלי. בפרט, אם קיים גם לוקאלי וגם גלובלי עם השם symbol_name – מדובר במקרה הראשון או האחרון, ואין להחזיר 2- אלא 1 או 4-.
  - **-3** – אם הקובץ אינו executable. שימו לב שבמקרה של "התנגשות שגיאות", השגיאה הזו עדיפה. למשל, אם הסמל קיים, אבל הקובץ אינו executable, יש להחזיר 3-.
  - **-4** – אם הסמל קיים, הוא גלובלי, אבל אינו מוגדר בקובץ הריצה. כלומר, אם מוגדר ב- shared object.
- ערך החזרה – במקרה של שגיאה (error_val<0), ערך החזרה של הפונקציה אינו מוגדר ולא ייבדק. במקרה שאין שגיאה, תוחזר הכתובת הווירטואלית אליה ייטען הסמל בתחילת ריצת קובץ הריצה.

הערות

1. שימו לב שפונקציית main נתונה. אין לשנות אותה!
2. שימו לב שקיבלתם קובץ עזר בשם elf64.h. השתמשו בו ובהגדרות שהוא מספק, על מנת להקל על הפרסור (parsing) של קובץ ה-ELF. אין לשנות את הקובץ, מכיוון שלא תגישו אותו לבסוף.
3. מלבד elf64.h, מותר להשתמש בספריות הסטנדרטיות של C ובספריית string.h, אך אין להשתמש בספריות ייעודיות כלשהן לניתוח קבצים ובנוסף אין להשתמש בכלים חיצוניים כדי לנתח את קובץ הריצה!
   פתרונות שיכללו שימוש בכלים חיצוניים (כדוגמת **readelf**) - יפסלו!
4. יש לסיים את התוכנית תמיד דרך ה-main בעזרת return 0. בפרט, אין לבצע exit בעצמכם בפונקציה.
5. אפשר להניח כי כל קריאות המערכת יצליחו ואין צורך לבדוק זאת בקוד המוגש. עם זאת, בשלב ה-debugging, זה כנראה יועיל לכם לבדוק זאת עבור עצמכם.

הגשה

עליכם להגיש את הקובץ **hw3_part1.c** בלבד. בפרט, לא להגיש את **elf64.h**.

חלק ב – Linker scripts

מבוא

חלק זה בתרגיל דורש היכרות עם תרגול 9. הסתכלו על התיעוד של ld, הידוע בשמו כ-GNU Linker, והתמקדו בחלק על Linker Scripts, בו ניעזר בחלק זה של תרגיל הבית.
מה זה Linker Scripts? tl;dr – מדובר בקובץ הגדרות ללינקר, בעזרתו שולטים על הדרך בה הוא מבצע את הקישור ומייצר את קובץ הריצה (כולל הגדרות הטעינה). ל-GNU Linker יש קובץ default בו הוא משתמש בריצה רגילה, אך ניתן להעביר לו קובץ אחר, עם הגדרות אחרות, כך שהקישור יתבצע כפי שאנחנו רוצים (העברה של קובץ אחר מבטלת את השימוש בקובץ ה-default). זה חלק ממה שנעשה בתרגיל.
איך עושים זאת? מייצרים קובץ 'ld.' ומכניסים אותו להרצת הלינקר עם הדגל T-, כפי שתראו בדוגמאות בהמשך.

כתיבת ה-Script

ה-Script שלכם, שמייצר קובץ ריצה, צריך לבצע את הדברים הבאים:

1. קובץ הריצה צריך להתחיל את ריצתו בסמל _hw3_unicorn.

   הערה: ניתן להניח שהסמל הזה יהיה קיים באחד מקבצי הקלט לקשר הסטטי, אין צורך לבדוק זאת.

2. קובץ הריצה צריך להכיל שלושה סגמנטים:
   I. סגמנט שמכיל את text section ובעל הרשאות כתיבה והרצה. הסגמנט ייטען לכתובת 0x400000.
   II. סגמנט שמכיל את data section ואחריו את bss section (כלומר data בכתובות נמוכות יותר), וסגמנט זה בעל הרשאות קריאה וכתיבה בלבד. הסגמנט ייטען לכתובת 0x60000.
   III. סגמנט שמכיל את rodata section וסגמנט זה בעל הרשאות קריאה והרצה בלבד. הסגמנט ייטען לכתובת 0x80000000.

   הערות:
   - ניתן להניח שכל section מהארבעה שצוינו כאן יהיה קיים לפחות באחד מקבצי הקלט.
   - יש לוודא שאם section לא קיים באחד מקבצי הקלט, זה לא יפריע ל-linker script לעבוד בצורה תקינה.

3. טבלת הסמלים של קובץ הריצה שלכם צריכה להכיל 2 סמלים חיצוניים (Ndx=UND):
   I. purple
   II. white

הגשה

לבסוף עליכם להגיש רק הקובץ `hw3_part2.ld`.

חלק ג' - הוראות הגשה לתרגיל בית רטוב 3

אם הגעתם לכאן, זו בהחלט סיבה לחגיגה. אך בבקשה, לא לנוח על זרי הדפנה ולתת את הפוש האחרון אל עבר ההגשה. חבל מאוד שתצטרכו להתעסק בעוד מספר שבועות מעכשיו בערעורים, רק על הגשת הקבצים לא כפי שנתבקשתם. אז קראו בעיון ושימו לב שאתם מגישים את כל מה שצריך ורק את מה שצריך.

עליכם להגיש את הקבצים בתוך zip אחד:

`hw3_wet.zip`

בתוך קובץ zip זה יהיו 2 תיקיות:

- part1
- part2

ובתוך כל תיקייה יהיו הקבצים הבאים (מחולק לפי תיקיות):

- `part1:`
  - `hw3_part1.c`
- `part2:`
  - `hw3_part2.ld`

בהצלחה!!!