

תרגיל בית 4

תיאור התרגיל

בבקשה תקראו את הסעיפים הבאים תוך כדי סקירה של קוד השלד שמצורף לתרגיל. תיאור דרישות התרגיל מתייחס לשמות משתנים שנמצאים בקוד המצורף.

בתרגיל זה נממש תוכנית המדמה את תהליך התרגום של כתובת וירטואלית (Virtual/Logical Address) לכתובת פיזית (Physical Address) ע"י שימוש בטבלת מיפוי לינארית שטוחה. אנחנו נניח כי גודל הטבלה הוא 256 דפים, וכי כל דף הינו בגודל של 256 בתים (bytes).

כמו כן, אנו נממש מנגנון TLB פשוט, אשר ישמור תרגומים שנעשו בעבר. אפשר להניח כי טבלת ה-TLB תקפה לאורך כל ריצת התוכנית. בנוסף, יש לשים לב כי טבלת ה-TLB היא בעלת גודל סופי, ולכן הכתיבה אליה תתבצע באופן ציקלי (cyclic) – כלומר, לאחר הכתיבה לתא האחרון בטבלה, נחזור ונדרוס את התא הראשון בטבלה בפעם הבאה שנכתוב לטבלה.

התוכנית מקבלת שני קבצים:

1. BACKING_STORE.bin: קובץ זה הינו קובץ בינארי שנרצה לטעון לזכרון הפיזי main_memory המדומה (לצורך התרגיל), ולקרוא ממנו ערכים.

2. addresses.txt: קובץ זה הינו קובץ טקסט אשר מכיל כתובות וירטואליות לתרגום.

המשתמש שמריץ את התוכנית, מספק לה את שני הקבצים הנ"ל, ועל התוכנית לקרוא כתובות וירטואליות זו אחר זו מקובץ הכתובות, ועבור כל כתובת logical_address לבצע את הפעולות הבאות:

1. לקדם את total_addresses ב-1 (אנו מונים את מספר הכתובות שאנו נדרשים למפות).
2. לחשב את logical_page באמצעות logical_address.
3. לבדוק האם logical_page נמצא ב-TLB. אם כן, אז יש להשתמש ב-physical_page שנתון ע"י ה-TLB, ולקדם את tlb_hits ב-1.
4. אם הדף הלוגי לא נמצא ב-TLB, אז נבצע את הפעולות הבאות:
 - a. נבדוק האם logical_page ממופה ע"י טבלת הדפים pagetable. אם הדף אינו ממופה ע"י הטבלה, אז משמעות הדבר כי ספגנו page-fault exception, ולכן, נבצע את הפעולות הבאות:
 - i. נקדם את page_faults ב-1.
 - ii. נגדיר את free_page להיות ה-physical_page שמתאים ל-logical_page. נקדם את free_page ב-1 (אנו מניחים שהדף הפיזי ה-0 הוא הדף הראשון שמוקצה. הדף הפיזי ה-1 יהיה הדף הבא שיוקצה, וכן הלאה...).
 - iii. נעתיק את מקטע הזכרון שמתאים ל-logical_page ב-BACKING_STORE, למקטע הזכרון שמתאים ל-physical_page ב-main_memory.
 - iv. נעדכן את ה-pagetable, כך ש-logical_page ממופה ל-physical_page.
 - v. נקדם את tlbindex ב-1, ונדרוס את התא במיקום tlbindex (מודולו TLB_SIZE) עם הצמד logical_page, physical_page.
 5. נעתיק את ה-byte שמתאים לכתובת הפיזית המלאה מ-main_memory למשתנה value.

קומפילציה, הרצה ובדיקה

בקובץ ה-ZIP של התרגיל מסופקים לכם הקבצים הבאים:

- 1. virtmem_skeleton.c
- 2. BACKING_STORE.bin
- 3. addresses.txt

לנוחיותכם, שלד התרגיל כבר מכיל מימוש מינימלי, ועליכם רק להשלים את קטעי הקוד החסרים. אנא קראו את הקוד המצורף, והבינו כיצד לממש את האלגוריתם שמתואר לעיל.

יש להדר את התוכנית באמצעות הפקודה הבאה:

```
gcc -Werror -std=c99 virtmem.c -o virtmem
```

הגשה:

ההגשה הינה אלקטרונית דרך Moodle. עקבו אחר השלבים הבאים:

1. עליכם ליצור קובץ zip (השתמשו ב-zip או gzip בלבד) בשם hw4_id1_id2 כאשר id1, id2 מייצגים את מספרי תעודות הזהות של המגישים.

2. תכולת קובץ ה zip צריכה להיות התכולה הבאה (ללא תתי ספריות!):

- o virtmem.c
- o קובץ בשם submitters.txt שמכיל את מספרי הזהות והשמות של מגישי התרגיל מופרדים על ידי פסיק במבנה הבא (לדוגמה):

```
Bill Gates,bill@microsoft.com,123456789  
Linus Torvalds,linus@gmail.com,234567890
```

3. את קובץ ה-zip יש ליצור ע"י הרצת הפקודה הבאה:

```
zip hw4_id1_id2.zip virtmem.c submitters.txt
```

הגישו את קובץ ה-zip דרך Moodle