מערכות הפעלה – תרגיל 4

לתרגיל זה 2 חלקים:

א. כתיבת מנהל התקן המדמה רכיב הצפנה.

ב. שאלות מילוליות.

בסוף התרגיל מפורטות הוראות ההגשה והנחיות כלליות. יש לקרוא אותן בעיון ולהקפיד על כל ההנחיות!

חלק א

עליכם לממש מנהל התקן בקבצים בשם **encdev.h** ,**encdev.c**, אשר מבצע הצפנה ופענוח על נתונים הנכתבים לרכיב המתאים.

ההצפנה היא צופן קיסר, כך שלכל בית שנכתב לרכיב מתווסף מפתח ההצפנה ובקריאה נחסיר את מפתח ההצפנה (כדי לקבל חזרה את הנתון המקורי). הערך שמוסיפים ומחסירים הוא מפתח ההצפנה.

את הרכיבים נייצג ע"י מערכים בזיכרון של מודול הגרעין. המודול מקבל 2 פרמטרים: פרמטר ראשון בשם size מסוג מספר (int) המציין את הגודל של כל מערך, ופרמטר שני בשם count המייצג את כמות הרכיבים הנתמכים. באתחול ההתקן יש להקצות דינמית count מערכים בגודל size כל אחד, ולדאוג לשחרור זיכרון תקין בסיום.

מנהל ההתקן יתמוך ב-count רכיבים לפי מספר ה-minor. כלומר, קובץ התקן עם minor של 0 ישתמש בנתונים במערך 0, וכן הלאה. אם נפתח קובץ התקן עם minor מחוץ לטווח, על פעולת הפתיחה להיכשל.

כדי לקבל את מספר ה-minor של קובץ ההתקן שנפתח, יש להשתמש במאקרו: **MINOR(inode->i_rdev)**.

בכל פתיחה של קובץ התקן יאותחל מפתח הצפנה עם ערך 0 עבור הגישה הנוכחית לקובץ ההתקן. יש לאפשר למשתמש לשנות את מפתח ההצפנה באמצעות פקודת IOCTL עם מזהה פקודה הנתון בקבוע **IOCTL_SET_KEY**. מפתח ההצפנה הוא בית אחד (בין 0 ל-255).

יש לממש פעולות קריאה, כתיבה, ו-seek, ויש להתייחס להיסט בכל פעולה ולעדכן אותו בהתאם. במידה ובוצעה פעולה קריאה כאשר ההיסט הוא size (או יותר) יש להחזיר EOF, בפעולת כתיבה יש להחזיר שגיאה. בכל מקרה אחר יש לקרוא או לכתוב כמה נתונים שמתאפשר (ובהתאם לבקשת המשתמש), ולהחזיר ערך מתאים. ניתן לממש פעולות נוספות לפי הצורך (open, release). יש לזכור ולדאוג להצפין ולפענח את הנתונים בפעולות כתיבה וקריאה, בהתאמה.

בכל פעולה של מנהל ההתקן ומודול הגרעין יש להדפיס הדפסה אחת ברורה ללוג של הקרנל (גם במקרה של שגיאה).

יחד עם מודול הגרעין יש לצרף גם קובץ Makefile מתאים אשר מקמפל אותו, וקובץ סקריפט בשם **encdev.sh** אשר מקבל ארגומנט יחיד ומבצע את הפעולות הבאות (לפי הסדר):

- מעתיק את קבצי הקוד של המודול יחד עם ה-Makefile אל התיקייה הנתונה בארגומנט הראשון.
- מקמפל את המודול וטוען אותו אל מערכת ההפעלה (עם size=1024, count=10).
- מוחק את כל הקבצים הזמניים שנוצרו: `sudo make clean`.
- מוחק את 3 הקבצים שהועתקו אל תיקיית היעד (את העותקים, הקבצים המקוריים לא נמחקים).

ניתן להניח שמתקבל ארגומנט יחיד והארגומנט תקין, ואת קובץ הסקריפט מריצים עם sudo.

מערכות הפעלה – תרגיל 4

<u>חלק ב</u>

נתון דיסק קשיח עם הנתונים הבאים:

- מהירות סיבוב: 3,000 RPM. גודל סקטור: 512B. גודל בלוק: 4KB.
- 6 משטחים (3 platters דו-צדדיים).
- צפיפות הסקטורים בכל רצועה היא בהתאם ל-4 אזורים: A, B, C, D. בכל אזור 100 רצועות, והאזורים מסודרים בדיסק מבחוץ פנימה – כאשר A הוא החיצוני ביותר ו-D הפנימי ביותר.
- באזור A יש 1200 סקטורים לרצועה, באזור B יש 800 סקטורים לרצועה, באזור C יש 600 סקטורים לרצועה, באזור D יש 400 סקטורים לרצועה.
- זמן ה-seek הממוצע הוא 20ms, בין שתי רצועות סמוכות הוא 2ms, והמקסימלי הוא 40ms.
- נתון שיפור track skew: בפעולה על סקטורים עוקבים ברצועות סמוכות אין המתנה לסיבוב.

1. מה שטח האחסון הכולל בדיסק **בבלוקים** (כמה בלוקים יש בדיסק כולו)?

2. קוראים תוכן של קובץ בדיסק בגודל 100 בלוקים, אשר מפוצל על פני שתי רצועות באותו משטח (בתוך כל רצועה המידע יושב באופן רציף). מדדו וניטרו פעולה זו וגילו שמתבצע, לפי הסדר:

a. השהיה מסוימת למשך זמן לא ידוע.
b. העברת 300KB של נתונים במשך 10ms.
c. השהיה של 2ms.
d. העברת שאר המידע של הקובץ (100KB) ברצף אחד, ללא השהיות נוספות.

מה הזמן **המקסימלי** שיתכן שנמשכה העברת המידע בשלב האחרון?

את התשובות יש להגיש במסמך **PDF (בלבד!)** בשם `myanswers.pdf`. יש לפרט את כל החישובים ולהסביר באופן ברור כל שלב בחישוב. נא להקליד את התשובות ולא לצרף צילום/כתב יד. תשובות שאינן מוקלדות לא יתקבלו!

מערכות הפעלה – תרגיל 4

<u>הוראות הגשה:</u>

- יש להגיש את התרגיל בקובץ tar.gz **יחיד**. שם הקובץ הוא ת"ז בלבד (9 ספרות) וסיומת.
לדוגמה, אם הת"ז הוא 123456789 אז שם הקובץ להגשה יהיה **123456789.tar.gz**
- הקובץ יכיל את 4 קבצי התרגיל **בלבד**, ללא תיקיות או קבצים נסתרים.
את הקובץ יש ליצור ב-VM באמצעות הפקודה הבאה:
```
tar czf <yourid>.tar.gz encdev.c encdev.h Makefile encdev.sh myanswers.pdf
```
- מצורף לתרגיל סקריפט פייתון (בשם checksub.py). יש להריץ אותו עם שם קובץ ההגשה כפרמטר, לבדיקת תקינות ההגשה. **לא יתקבלו ערעורים על הגשה לא תקינה**.

<u>הנחיות כלליות:</u>

- ההגשה היא **ביחידים**. כל דמיון בין הגשות **ולו הקטן ביותר** יועבר <u>מיידית</u> לטיפול ועדת המשמעת של המכללה ללא אזהרה או אפשרות ערעור. מקורות אונליין (כולל LLM כמו ChatGPT) שקולים להעתקה. פתרו את התרגיל **<u>באופן עצמאי לחלוטין</u>** והימנעו מבעיות.
- בכל ערעור על קוד יש לצרף תמונת מסך של הטרמינל עם הפלט של פקודת `hostnamectl`, ביצוע make לקוד, והרצה של הקוד עם פלט תקין – מתוך VM על גבי vagrant כפי שנלמד.
מי שלא פועל לפי ההנחיות (מערכת הפעלה שונה, או ללא vagrant) – **<u>מוותר על הזכות לערער</u>**.
- **הגשות באיחור** יתקבלו עם הפחתה של 10 נק' לכל יום איחור מעבר למועד ההגשה, אשר מסתיים בדיוק **<u>בשעה 23:00</u>**. דאגו להגיש מספיק זמן מראש, לא יתקבל אישור על פספוס ברגע האחרון.
במקרה של מחלה, מילואים, או כל סיבה מוצדקת אחרת, יש לשלוח אישור אל המרצה ***<u>לפחות 24 שעות</u>*** לפני ההגשה.