**מסמך אפיון- רטוב 2**

יצירת חוט +ארגומנטים

פעולות לביצוע

ATM

מידע + חיווי על הצלחת פעולה

סיום חוט

Account

main

סיום חוט

גביית עמלה

יצירת חוט +ארגומנטים

האם VIP

Bank

במודול הmain יוצרים את 2 החוטים המרכזיים שמנהלים את הפעילות של הבנק. אל הכספומטים מועברים הארגומנטים שניתנו מהמשתמש – מס' כספומטים וקבצי פעולות שעליהם לבצע. אל הmain חוזר מידע על סיום פעילות החוט על מנת לשחרר את המשאבים והחוטים בצורה מבוקרת.

במודול ATM מייצרים חוטים לכל כספומט ולכל חוט נותנים את קובץ הפעולות המתאים לו. מתבצעת קריאה של הקובץ שורה-שורה והפו' המתאימה נקראת. בסיום פעולות כספומט יש החזרת מידע על סיום פעילות של החוט.

במודול Bank יוצרים חוט של גביית עמלות וחוט של הדפסה. יש שימוש בפו' ממודול החשבון בפעולת הגבייה וההדפסה.

מודול Account מקבל ארגומנטים לריצת המתודות מהכספומט והבנק ועושה את הפעלות הנדרשות ומחזיר מידע בהתאם.

מבנה נתונים:

השתמשנו ב STL Map על מנת להחזיק את החשבונות. המפתח הוא מספר החשבון ושאר המידע: סיסמא, יתרה, הם הערכים המוחזקים במבנה הנתונים. המבנה היה גלובלי על מנת לאפשר גישה לכל הקבצים אל החשבונות.

יצרנו class Account שבו יש מתודות שמאפשרות את הפעולה של הכספומט כמו גם מתודות שמחזירות מידע על המצב הנוכחי של משתני הclass.

מימוש מקביליות

בתחילת העבודה ניסינו לחשוב על מקביליות מקסימלית ואלה הקווים שהנחו אותנו:

* לבדוק מה הם המשאבים המשותפים
* הפרדה מרבית בין המשאבים המשותפים
* מנעול שונה לכל משאב.

מתוך הנחות הללו הגענו להפרדה הבאה:

ראשית הפרדנו בין החשבונות השונים ובתוך החשבון ישנם שני משאבים משותפים שגם ביניהם הפרדנו:

א. היתרה

ב. האם הוא vip

בנוסף יש צורך לנעול כתיבה למסמך log שכן מערכת ההפעלה אינה מגינה על מצב של מספר כותבים ומנעול נוסף על מנת לאפשר נעילה כללית שלכלל הפונקציות למפת החשבונות.

מתוך דרישות התרגיל נאלצנו להוריד ברמת המקביליות:

* מתוך הדרישה של נכונות לקיחת העמלה איחדנו את המנעולים של vip ויתרה למנעול אחד לחשבון.
* מתוך הדרישה של הצורך להקפיא כתיבה לכלל החשבונות בעת הדפסת מצב הבנק הוספנו מנעול המפריד בין מצב הדפסת בנק (נעילת כלל החשבונות במנעול יחיד) לבין שינוים בחשבונות.
* מתוך הדרישה של הדפסה סדרתית של עבור לקיחת העמלה השתמשנו בשילוב בין שני מנעולים, מנעול ההדפסה ומנעול הגישה. מה שעוצר כתיבות חדשות לכתוב לחשבונות ופעולות חדשות לפעול בכספומטים.

מכאן קיבלנו ארבעה סוגי מנעולים שבכל אחד מהם כתבנו אלגוריתם נעילה שונה.

**1. נעילת חשבון בודד**

במקרה זה אנחנו רואים שיש יותר פעולות כותבות (לקיחת עמלה, שינוי סטטוס Vip, הפקדה, משיכה והעברה) מאשר פקודות קוראות (מצב החשבון, הדפסת הבנק) ולכן התאמנו את האלגוריתם שלנו להעדפה של כותבים על קוראים. מתוך הצורך הזה מצאנו אלגוריתם כזה המשתמש בארבעה מנעולים ושני סופרים:

1. מנעול כותבים(write) – מנעול המשותף לכלל הכותבים הנדרש לצורך עדכון סופר הכותבים
2. מנעול קוראים (read)– מנעול המשותף לכלל הקוראים הנדרש לצורך עדכון סופר הקוראים
3. מנעול רשות לקוראים(readtry) – מנעול שבעזרתו יצרנו את העדפה לכותבים (יורחב בהמשך)
4. מנעול המשאב (resource)– חוסם כתיבה וחוסם שני כותבים בו זמנית

אלגוריתם הכתיבה:

Lock(&write)

Write\_counter++;

If (write\_counter == 1) // first writer to write)

Lock (&**readtry**); //locking the readers from entering the **counter**

Unlock (&write);

Lock(&resource); //locking other writers to enter

Do\_write();

Unlock(&resource);

Lock (&write)

Write\_counter- - ;

If (Write\_counter == 0) // no more writers

Unlock(&**readtry**)

Unlock(&write)

ניתן לראות שכמו באלגוריתם הקוראים -כותבים שנלמד בכיתה אנחנו משתמשים בסופר על מנת לאפשר כניסות מרובות (כמו בקוראים) אבל חוסמים קוראים מלהיכנס בכלל.

אלגוריתם הקריאה:

Lock(&readtry)

Lock(&read)

Read\_counter++;

If (Read\_counter == 1) // first writer to read

Lock (&**resource**); //locking from writers to write

Unlock (&read);

Unlock (&readtry);

Do\_read();

Lock (&read)

Read\_counter- - ;

If (Read\_counter == 0) // no more writers

Unlock(&**resource**)

Unlock(&read)

המנגנון של האלגוריתם מתבסס על המצב שבו יש מספר קוראים שנמצאים בקריאה ואחד מהכותבים הגיע אז בזמן הנעילה הוא נועל קוראים נוספים מלהיכנס והוא מנסה לנעול את המשאב (שכרגע נעול ע"י הקוראים).

במצב זה לא יכנסו קוראים נוספים וכאשר כל הקוראים יסיימו את הקריאה הם יתנו לכותב להתחיל לכתוב.

**2. נעילת הבנק משינויים**

במקרה זה אנחנו רוצים להפריד בין שתי מצבים. בין פעולות בנק (הדפסה ולקיחת עמלה) לבין פעולות שינויים המבצעים הכספומטים.

לכן נדרשנו להביא מנגנון נוסף הנועל את כלל החשבונות מ**שינויים** ע"י הכספומטים.

למנגנון זה השתמשנו באותו אלגוריתם של קוראים-כותבים אך החלפנו ביניהם כלומר הקוראים הם הפעולות הרבות שעושים הכספומטים על החשבונות השונים (קוראים רבים) והכותבים הם פעולות הבנק (פעולות מעטות)

מנגנון הנעילה הזה דרש מאתנו שני מנעולים וסופר.

**3. נעילת הגישה למפת החשבונות**

נעילה זה באה לוודא שבכל עת שאנחנו באים לגשת למפה לא נוספו אנשים חדשים. כלומר כאשר פעולה (הוספה איש חדש או שאר הפעולות גם כן) רוצות לבדוק אם משתמש כבר נמצא אז המנעול הזה מאפשר סינכרון בין הוספת המשתמש החדש לשאר הפעולות שבאות לגשת למפה.

נעילת מנעול זה באה כפעולה ראשונה של כלל פעולות הכספומטים ולפני בדיקת הימצאות החשבון במפה.

בנוסף השתמשנו במנעול זה על מנת ליצור לקיחת עמלה כבלוק רציף ולמנוע מפעולות אחרות להיכנס ולכתוב לlog בזמן שלקיחת העמלה מתבצעת.

**4. נעילת קובץ הlog**

במקרה זה יש לנו כותבים רבים (אין קוראים) שמנסים לכתוב למשאב משותף יחיד. השתמשנו במנעול בודד הנותן אפשרות כתיבה למי שמצליח לנעול אותו.