

תרגיל רטוב מספר 2

שם: נדב שלו

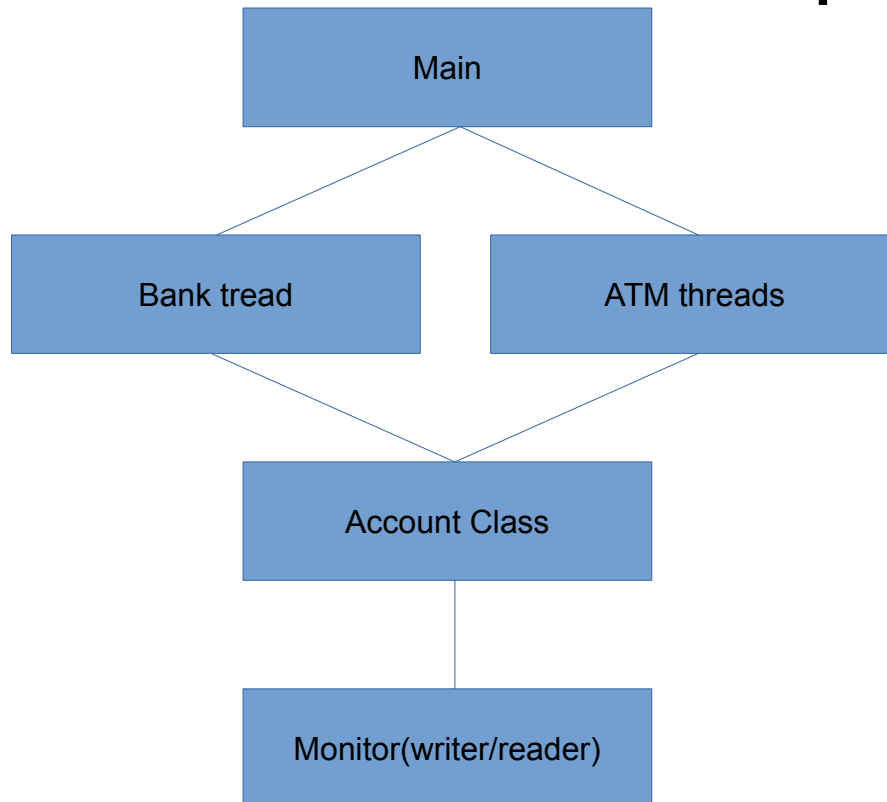
תז: 206203325

שם: חנן גביר

תז: 318163243

תאריך: 20.12.19

תרשים הפתרון



מבני נתונים

1. `list <Account*> accounts`
רשימה שמחזיקה את כל החשבונות במערכת
2. `stringstream strLog`
BUFFER שלתוכו יכנסו תוצאות הפקודות השונות (והוא יודפס לקובץ באופן מחזורי)
3. `pthread_mutex_t logLock, accountListLock, bankCountLock`
מנעולים שונים כמתואר בהמשך

תיאור הפיתרון

1. מימשנו מוניטור אבסטרקטי של קוראים כותבים. כך אנו יכולים להתחיל ולסיים קריאה וכתבייה מבלי לתפוס את המנעול מעבר לנדרש.
2. בעזרת מחלקת המוניטור מימשנו מחלקת חשבונות (Account) ששומרת אובייקט של המוניטור כמשתנה פנימי. כך כאשר מיצרים חשבון חדש הוא מקבל אובייקט נפרד של מוניטור.
3. הפונקצייה הראשית מחזיקה רשימה (List) של חשבונות וכן מקבלת מהמשתמש את מספר ושמות הקבצים של הכספומטים שהוא רוצה לאתחל. מרציה כל ATM בחוט נפרד ומעבירה לו כפרמטר את נתיב הקובץ של הפקודות וכן ID שונה לכל אחד. מאתחלת חוט שיריץ את BANK. מעבירה לו מצביע למשתנה שמחזיק את סך הכסף שהבנק הרוויח. חוט נוסף מריץ את ההדפסה כל חצי שנייה
4. חוט ATM:
 1. קורא את הקובץ ומריץ אותו פקודה פקודה בהתאם לדרישות.
 2. משתמש במחלקת חשבונות למימוש המקביליות
 3. לפני הכנסה של תוצאה ל log של הפקודות – משתמש במנעול על מנת להבטיח הכנסה סידרתית נכונה ללוג (למנוע מצב ששתי פקודות יוכנסו ללוג ויתערבבו בהדפסה אליו)
 4. כאשר מכניס חשבון חדש לרשימה – נועל אותה לפני כן על מנת למדוע הדפסה של רשימה לא מעודכנת למך
5. חוט BANK:
 1. מממש מנגנון עמלות כנדרש בעזרת שימוש במחלקת חשבונות
 2. מעדכן משתנה שמכיל את הכסף שהבנק הרוויח – נועל אותו לפני כן על מנת להבטיח נכונות
6. חוט הדפסה:
 1. עובר על רשימת החשבונות ומכניס את ההדפסה שלה ל BUFFER - ראשית נועל אותה להבטחת נכונות
 2. מוחק את המסך ומדפיס את כל ה BUFFER בבת אחת – זאת על מנת למנוע ריצודים מיותרים.
 3. שומר את ה LOG לתוך הקובץ – ראשית נוכל את ה LOG להבטחת נכונות.