**מבני נתונים עבודה 2**

מגישות:

* יובל הרינג -204537955
* מאיה אוקסנהנדלר – 302257563

המחלקה Page- המחלקה מייצגת אובייקט מסוג דף אשר נשמר בזכרון הראשי.

שדות:

* + Key- מזהה מספרי חח"ע עבור הדף. מייצג את המיקום של הדף בזכרון המשני.
  + pageContent- תוכן הדף, מיוצג ע"י מחרוזת.
  + Prev- מייצג את המפתח עבור הדף הקודם שנערך. במידה וזהו הדף הראשון שנערך הערך יהיה 1-.
  + Next- מייצג את המפתח עבור הדף הבא שנערך. במידה וזהו הדף האחרון שנערך הערך יהיה 1-.

(הערה: במשתנים prev ו-next נעשה שימוש רק כאשר עובדים בשיטת lru לכן במצב של fifo הם מאותחלים להיות 1- ולא משתנים).

פונקציות:

* בנאי- מאתחל את המפתח לפי המפתח שהתקבל, מאתחל את תוכן הדף לפי התוכן שהתקבל ואת המצביעים להיות 1-. **סיבוכיות-O(1)**
* editIndexes- משנה את המצביעים של הדף ושל הסמוכים לו לפי המיקום של הדף בתור. מתבצעת קריאה רק כאשר עובדים בשיטת lru. **סיבוכיות-O(1).**
* getPageContent- מחזירה את תוכן הדף. **סיבוכיות- O(1).**

המחלקה PageQueueFIFO- המחלקה מממשת תור של דפים, כך שהאיבר הראשון שהוכנס הוא הראשון לצאת

שדות:

* head- משתנה מספרי המחזיק את המקום אליו נכניס את הדף הבא שיגיע לתור.
* numOfElements- משתנה מספרי המחזיק את מספר הדפים בתור בכל רגע.
* Pages- מערך של דפים בגודל n המייצג את התור. **סיבוכיות זכרון -O(n)**

פונקציות:

* בנאי- מאתחל את השדות-head, numOfElements=0 ואת גודל המערך להיות גודל הזיכרון הראשי. **סיבוכיות- O(1).**
* Enqueue- מכניסה את האיבר שהתקבל למקום המתאים בתור. עורכת את head ו- numOfElements בהתאם. **סיבוכיות- O(1).**
* Dequeue- מוציאה את האיבר המתאים מהתור. עורכת את numOfElements ומחזירה את האיבר שיצא. **סיבוכיות- O(1).**
* Get- מקבלת אינדקס ומחזירה את הדף המתאים. **סיבוכיות- O(1).**
* getHead- מחזירה את head. **סיבוכיות- O(1).**

המחלקה PagesQueueLRU- המחלקה מממשת תור של דפים כך שהדף הראשון לצאת הוא הדף שזמן העריכה שלו הוא הרחוק ביותר.

שדות:

* toDequeue- ערך מספרי המחזיק את המפתח של הדף הבא שצריך לצאת מהתור.
* Last- ערך מספרי המחזיק את המפתח של הדף האחרון שנערך.
* numOfElements- מספר הדפים הקיימים בתור בכל רגע.
* Pages- מערך של דפים בגודל הזכרון המשני m אשר מייצג את התור, לאחר סיום כל פעולה קיימים בתור n דפים המייצגים את הדפים בזכרון המשני, כאשר בשאר המקומות במערך נמצא הערך null. **סיבוכיות זכרון O(n\*pageSize+(m-n))**

פונקציות:

* בנאי- מאתחל את גודל המערך להיות גודל הזכרון המשני, numOfElements=0. **סיבוכיות- O(1).**
* setSpecials- מאתחל את toDequeue ואת last. **סיבוכיות- O(1).**
* Enqueue- מכניס איבר למקומות בתור לפי המפתח. עורך את האינדקסים של הדף ושל הדפים הסמוכים לו בהתאם. **סיבוכיות- O(1).**
* Dequeue- מוציא מהתור את האיבר במקום ה toDequeue עורך את האינדקסים של הדף ושל הדפים הסמוכים לו בהתאם. **סיבוכיות- O(1).**
* getPage- מקבל מפתח ומחזיר את הדף המתאים. **סיבוכיות- O(1).**
* setToDequeue- עורך את האינדקס של toDequeue. **סיבוכיות- O(1).**

המחלקה MemoryManagementSystem- מחלקה המנהלת את הזכרון הראשי.

שדות:

* secondaryMemory- מערך של מחרוזות, מייצג את הזכרון המשני.
* useLRU- משתנה בוליאני המגדיר באיזו שיטה להשתמש
* FIFOPagesQueue- תור בשיטת FIFO מאותחל רק במידה ו-useLRU=false.
* locationInMain- מערך מספרים בגודל m אשר שומר עבור כל מפתח איפה הוא נמצא בזכרון הראשי – בשימוש רק במידה וuseLRU=false.
* LRUPagesQueue- תור בשיטת LRU מאותחל רק במידה ו- useLRU=true.

פונקציות:

* בנאי- מאתחל את התור המתאים בהתאם לערך בuseLRU. **סיבוכיות O(m)**.
* Read- מקבלת אינדקס ומחזירה את תוכן הדף שזהו המפתח שלו. במידה ודף לא נמצא בזכרון הראשי מתבצע חילוף באמצעות התור שנבחר. **סיבוכיות- O(1).**
* Write- מקבלת אינדקס ותו, מכניסה את התו בסוף התוכן של הדף שזו המפתח שלו. במידה ודף לא נמצא בזכרון הראשי מתבצע חילוף באמצעות התור שנבחר. **סיבוכיות- O(1).**
* toString- מחזירה מחרוזת המייצגת את המצב של הזכרון המשני.
* getSecondary- מאפשרת גישה ישירה לכל ערך בזכרון המשני. **סיבוכיות- O(1).**
* isExist- מקבלת אינדקס ובודקת האם הדף שזהו המפתח שלו קיים בזכרון הראשי.