

מבני נתונים – פרויקט מס. 2.

ערימת פיבונאצ'י

מבוא

במטלה זו אתם מתבקשים לממש ערימת פיבונאצ'י, כפי שנלמדה בכיתה.

הדרישות

עליכם לממש ערימת פיבונאצ'י (Fibonacci Heap), לפי ההגדרות שניתנו בכיתה. תוכלו למצוא את הפרטים במצגת שבמודל וב-Cormen.

בתרגיל זה נניח שהאיברים בערימה הם תמיד מספרים שלמים שונים זה מזה. המימוש צריך להיות מבוסס על קובץ השלד המופיע באתר הקורס. הפעולות שמופיעות בקובץ:

isEmpty(): הפונקציה מחזירה ערך TRUE אם ורק אם הערימה ריקה.

insert(int i): הפונקציה יוצרת צומת מסוג HeapNode שמכיל את המפתח i ומכניסה אותו לערימה. פעולה זו מחזירה את הצומת שנוצר ושמייל את המפתח i . על מנת לבצע את פעולות decreaseKey, delete בהמשך, יש להעביר אליהן צומת זה שמוחזר מפעולת ה insert.

deleteMin(): מחיקת הצומת שהמפתח שלו מינימלי מבין המפתחות שבערימה (אין צורך להחזיר אותו).

findMin(): מחזיר את הצומת (מטיפוס HeapNode) שהמפתח שלו מינימלי מבין המפתחות שבערימה.

meld(heap2): מיזוג הערימה עם ערימה נוספת heap2.

size(): הפונקציה מחזירה את מספר האיברים בערימה

countersRep(): הפונקציה מחזירה מערך מונים כך שבאינדקס i שמור כמה עצים יש בערימה שהסדר שלהם הוא i . כלומר, היא מחזירה מערך של integers, כך שלכל אינדקס i בין 0 עד הדרגה המקסימלית של עץ שקיימת בערימה, הערך שמוחזר במערך הוא מספר העצים שקיימים בערימה מסדר i . הערה: זו הפונקציה היחידה שאין צורך לבצע ביעילות. תממשו איך שהכי נוח לכם.

delete(HeapNode x): מחיקת הצומת x מהערימה.

decreaseKey(HeapNode x, int Δ): ערכו של המפתח של הצומת x יופחת ב Δ . כלומר, מתבצע $x.key \leftarrow x.key - \Delta$, וכמובן שיש לעדכן את מבנה הערימה בהתאם לשינוי זה (למשל לבצע Cascading Cuts במידת הצורך).

potential(): הפונקציה מחזירה את ערך הפוטנציאל הנוכחי של הערימה.

הפוטנציאל, כפי שהוגדר בשיעור, הינו
 $Potential = \#trees + 2 * \#marked$.

totalLinks(): פונקציה סטטית זו מחזירה את מספר כל פעולות הלינק שבוצעו מתחילת ריצת התוכנית. פעולת לינק הינה הפעולה שמקבלת שני עצים מאותו סדר ומחברת אותם.

totalCuts()

פונקציה סטטית זו מחזירה את מספר כל פעולות ה cut שבוצעו מתחילת ריצת התוכנית. פעולת cut קורת בזמן decreaseKey, כאשר מנתקים תת-עץ מהאבא שלו (כולל cascading cuts).

kMin(FibonacciHeap H, int k): פונקציה סטטית זו מקבלת עץ בינומי H (עם $2^{\deg(H)}$ צמתים) ומספר חיובי $k < size(H)$ ומחזירה מערך ממויין של k הצמתים הקטנים ב-H. על הפעולה לרוץ בסיבוכיות $O(k(\log k + \deg(H)))$.

בקובץ השלד מופיעים ה-header-ים של כל הפונקציות. המימוש יבוצע על ידי מילוי קובץ השלד. במידת הצורך ניתן להרחיב את המימוש (למשל להוסיף פונקציות עזר שלא מופיעות בשלד), אך אסור לשנות את ההגדרות של הפונקציות לעיל.

סיבוכיות

יש לתעד את סיבוכיות זמן הריצה (במקרה הגרוע) של כל פונקציה, כתלות במספר האיברים בעץ. עליכם להשיג סיבוכיות זמן ריצה זהה לזאת שנלמדה בכיתה עבור פעולות של מבנה הנתונים (זמני worst case ו-amortized צריכים להיות אלה שנלמדו בכיתה). לגבי פונקציות שלא נלמדו, עליכם להשיג זמן ריצה אסימפטוטי ב-WC טוב ביותר, כך שלא תפגעו בזמן ריצה WC או amortized של אלה שנלמדו.

תיעוד

קובץ המקור ייבדק גם באופן ידני. חשוב להקפיד על תיעוד לכל פונקציה, וכמות סבירה של הערות. הקוד צריך להיות קריא, בפרט הקפידו על בחירת שמות משתנים ועל אורך השורות. לקוד המקור יצורף מסמך תיעוד חיצוני. המסמך יכלול את תיאור המחלקה שמומשה, ואת תפקידו של כל חבר במחלקה. עבור כל מתודה במחלקה יש לפרט מה היא עושה, כיצד היא פועלת ומה סיבוכיות זמן הריצה שלה. בפרט, אם פונקציה קוראת לפונקציית עזר, יש להתייחס גם לפונקציית העזר בניתוח.

בדיקות

התרגילים ייבדקו באמצעות תוכנת טסטר שקוראת לפונקציות המפורטות מעלה, ומוודאת את נכונות התוצאות. קובץ הטסטר שלנו לא יפורסם לפני הבדיקות. עליכם לבדוק את המימוש בעצמיכם! בפרט, כדאי מאוד לממש טסטר, כדי לבדוק את תקינות ונכונות המימוש.

בקובץ שתגישו לא תהיה פונקציית main. אם הצלחתם לקמפל את הפרוייקט לבדו (ללא טסטר), זה סימן שמשהו לא נכון במימוש שלכם.

הקוד ייבדק על מחשבי בית הספר על גירסא Java8.

הנחיות להשמת סביבת העבודה בבית (ג'אוה+אקליפס):
<http://courses.cs.tau.ac.il/1415b/misc/workenv/pdf/software1/1415b/misc/workenv/>

מדריך (סעיפים 9-15):

<http://www.vogella.com/>

הנחיות לפתיחת חשבון מחשב, למי שמעוניינת לעבוד במעבדת בית הספר:

מדידות

כתבו תוכנית (לא להגיש אותה) שתפעיל את הפעולות שמימשתם, וענו בעזרתה על השאלות הבאות.

Sequence 1:

הרץ את סדרת הפעולות הבאות (הרצה חדשה עבור $m = 2^{10}, m = 2^{11}, m = 2^{12}$)

Insert(m), insert($m-1$), ..., insert(0)

delete-min()

decrease-key($m * (\sum_{k=1}^i 0.5^k) + 2$) for $i = 0, \dots, \log m - 2$

decrease-key($m-1$)

ניתן לבחור Δ שרירותי עבור פעולות decrease-key. בנוסף, שמרו מערך של מצביעים ל-heap nodes כדי שתוכלו להשתמש במתודה decreasekey.

מלאו את הטבלה: מה היה זמן הריצה שלקח להריץ את כל סדרת הפעולות הזו, כמה פעולות לינק בוצעו totalLinks וכמה פעולות חיתוך totalCuts וכן מה הפוטנציאל של המבנה נתונים בסוף הסדרה.

m	Run-Time (in milliseconds)	totalLinks	totalCuts	Potential
1024				
2048				
4096				

- מהו זמן הריצה האסימפטוטי של סדרת פעולה זו כפונקציה של m ?
- כמה פעולות Link וכמה פעולות cut מבוצעות, כפונקציה של m , במהלך סדרה זו של פעולות (יש לכתוב תשובה בצורה אסימפטוטית "Big O", אין צורך לבצע חישוב מדויק)?
- חשבו את עלות פעולת decrease-key היקרה ביותר במונחים של m (עבור חזקה כלשהי של 2). בדוק שהתשובות התיאורטיות (כפונקציה של m) תואמות לטבלה שקיבלת.

Sequence 2:

הרץ את סדרת הפעולות הבאות (הרצה חדשה עבור $m=1000$, $m=2000$, $m=3000$)

`Insert(m), insert(m-1), ..., insert(1)`

`deleteMin(), deleteMin(), ..., deleteMin()` (run delete min $m/2$ times).

מלא את הטבלה: מה היה זמן הריצה שלקח להריץ את כל סדרת הפעולות הזו, כמה פעולות לינק בוצעו `totalLinks` וכמה פעולות חיתוך `totalCuts` וכן מה הפוטנציאל של המבנה נתונים בסוף הסדרה.

M	Run-Time (in milliseconds)	totalLinks	totalCuts	Potential
1000				
2000				
3000				

א. מהו זמן הריצה האסימפטוטי של סדרת פעולה זו כפונקציה של m ?
ב. כמה פעולות Link וכמה פעולות cut מבוצעות, כפונקציה של m , במהלך סדרה זו של פעולות (יש לכתוב תשובה בצורה אסימפטוטית "Big O", אין צורך לבצע חישוב מדויק)?
ג. מה הפוטנציאל של המבנה כפונקציה של m בסוף ריצת סדרת פעולות זו? (יש לכתוב תשובה בצורה אסימפטוטית "Big O", אין צורך לבצע חישוב מדויק)?
ד. בדוק שהתשובות התיאורטיות (כפונקציה של m) תואמות לטבלה שקיבלת.

הגשה

הגשת התרגיל תתבצע באופן אלקטרוני באתר הקורס במודל.
הגשת התרגיל היא בזוגות בלבד!
כל זוג ייבחר נציג אחד ויעלה תחת שם המשתמש שלו את קבצי התרגיל למודל. על ההגשה לכלול שני קבצים: קובץ המקור (הרחבה של קובץ השלד שניתן), ומסמך תיעוד חיצוני, המכיל גם את תוצאות המדידות. את המסמך יש להגיש באחד הפורמטים הבאים: pdf, doc, docx, rtf, txt או pdf.
שמות הקבצים צריכים לכלול את שמות המשתמש האוניברסיטאיים של שני המגישים (לדוגמה, Heap_username1_username2.java). בתוכן הקבצים יש לציין את שמות המשתמש, תעודות הזהות ושמות המגישים (בכותרת המסמך ובשורת הערה בקובץ המקור).

הגשת שיעורי הבית באיחור - באישור מראש בלבד. הגשה באיחור ללא אישור תגרור הורדת נקודות מהציון או פסילה של התרגיל.
הגשת התרגיל היא חובה לשם קבלת ציון בקורס.

בהצלחה!