מבני נתונים 234218 תשע"ט

גיליון יבש מספר 3 – מעודכן לתאריך 09.01.2019

<u>מתרגל ממונה על התרגיל:</u> ניב גלעדי, giladiniv@cs.technion.ac.il

<u>תאריך ושעת הגשה:</u> 24.01.2018 בשעה 14:30.

<u>אופן ההגשה:</u> בזוגות. יורד ציון לתרגילים שיוגשו ביחידים בלי אישור מהמתרגל הממונה על התרגיל.

<u>הנחיות לפתרון:</u>

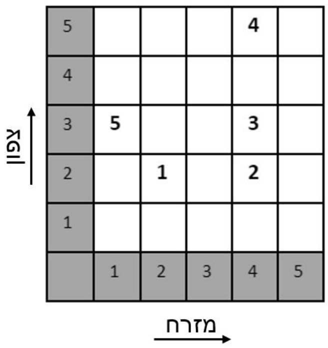
- כתבו בכתב ברור או הדפיסו את פתרונותיכם.

<u>זוג</u> שיגיש פתרון ש<u>כולו</u> (ולא רק חלקים ממנו) <u>מודפס ומוקלד</u> <u>משני צידי הדף</u>, יחד עם <u>דף שער מתאים</u> (דף שער לתרגיל יבש / דף שער להגשה מאוחרת, הנמצאים באתר הקורס), יקבל בונוס של 5 נקודות עד לציון מקסימלי של 100 נקודות.

- הקפידו לצרף את כל השאלות והסעיפים לפי הסדר! אי עמידה בכלל זה תגרור הורדה של ציון.
- הקפידו לכתוב את פתרונותיכם באופן מסודר ומובנה. התחילו מפתיח המתאר את תשובתכם בקצרה (עד 3 שורות) ולאחריו כתבו פירוט מלא של הפתרון. אי עמידה בכלל זה תגרור הורדה של 10% מהציון.
- אין צורך לפרט דברים שנלמדו בהרצאות או בתרגולים. מספיק לצטט או להפנות לחומר הלימוד. עם זאת, יש להוכיח כל טענה שלא נלמדה בהרצאה או בתרגול.
- יש לנתח סיבוכיות זמן ומקום של כל אלגוריתם ומבנה נתונים, אלא אם צוין אחרת.
- סיבוכיות זמן ומקום הן במקרה הגרוע, אלא אם צוין אחרת.
- תוכן ה-FAQ הינו מחייב, הקפידו להישאר מעודכנים.

שאלה 1 (20 נקודות)

UNION-FIND



בשאלה הבאה נתכנן מבנה נתונים לניהול תנועת רכבות בין n תחנות רכבת.
המיקום של כל תחנה יתואר ע"י 2 מספרים שלמים (x, y) בתחום $[1, n]$, ומכל תחנה ניתן להתקדם לכיוונים צפון ומזרח.
כל תחנה יכולה להיות פעילה או סגורה. בתחילת השאלה, כל תחנות הרכבת הן פעילות.
שימו לב: מספר התחנות n הוא גם חסם לקואורדינטות מיקומי התחנות.
דוגמה:
5 תחנות שנמצאות במיקומים $(2,2)$, $(2,4)$, $(3,4)$, $(5,4)$, $(3,1)$.

1. הציעו מימוש למבנה נתונים התומך בפעולות הבאות:

$Init((x_1, y_1)..(x_n, y_n))$ מקבלת n קואורדינטות של n התחנות, כאשר (x_i, y_i) הם הקואורדינטות של התחנה ה-i. לכל i מתקיים: $1 \le x_i, y_i \le n$.
<u>סיבוכיות זמן:</u> $O(n)$, כאשר n הוא מספר התחנות.

$DriveInDirection(i, d)$ מחזירה את התחנה הפעילה הבאה של רכבת שיוצאת מתחנה i בכיוון d, כאשר d יכול לקבל את אחד הערכים: $\{North, East\}$, ו-i הוא מספר תחנה. אם לא קיימת תחנה פעילה בכיוון d, יוחזר שזו התחנה האחרונה בכיוון זה. התחנה i יכולה להיות פעילה או סגורה.
<u>סיבוכיות זמן:</u> $O(log^* n)$ משוערך, כאשר n הוא מספר התחנות.

$CloseStation(i)$ התחנה ה-i נסגרת לשיפוצים. רכבות יכולות לעבור בתחנה הזו, אך לא לעצור בה או לשנות כיוון.
<u>סיבוכיות זמן:</u> $O(log^* n)$ משוערך, כאשר n הוא מספר התחנות.

סיבוכיות מקום: $O(n)$, כאשר n הוא מספר התחנות

2. מסלול חוקי של רכבת מוגדר כסדרה של תחנות פעילות $s_1 \to s_2 .. \to s_k$, כאשר לכל $i \in \{1, ..k-1\}$, s_{i+1} היא התחנה הפעילה הבאה מצפון או ממזרח ל-s_i.

לפי הדוגמה בסעיף א', המסלול $1 \to 2 \to 3 \to 4$ הוא חוקי כאשר כל התחנות פעילות, ולא חוקי כאשר תחנה 3 סגורה. המסלול $1 \to 2 \to 4$ חוקי לאחר סגירת תחנה 3. המסלול $3 \to 5$ אינו חוקי כלל, שכן רכבות יכולות להתקדם לכיוונים צפון ומזרח. אורך המסלול מוגדר כמספר התחנות הפעילות באותו מסלול.
הראו כיצד ניתן להוסיף את הפעולה הבאה למבנה הנתונים:

$MaxPath(start, end)$ מקבלת תחנת התחלה $start$ ותחנת סיום end, ומחזירה את המסלול הארוך ביותר שקיים בין התחנה $start$ ו-end.

ניתן להניח כי $start$ ו-end הן תחנות פעילות, ושקיים מסלול חוקי מ-$start$ ל-end.

מבני נתונים 234218 תשע"ט

גיליון יבש מספר 3 – מעודכן לתאריך 09.01.2019

<u>סיבוכיות זמן:</u> $O(nlog^*n)$ **במקרה הגרוע** כאשר n הוא מספר התחנות

שאלה 2 (20 נקודות)

בשאלה זו עליכם לממש מילון התומך בפעולות insert(x) ו- find(x) בלבד, כך ששתי הפעולות תרוצנה בזמן $O(1)$ בממוצע על הקלט (ללא "משוערך"), וסיבוכיות המקום של המבנה תהיה $O(n)$ כאשר n זה מספר האיברים במבנה בכל רגע נתון.

רמז: נסו לחשוב כיצד לחלק את הפעולה הכבדה הנדרשת לאורך הרבה פעולות אחרות.

שאלה 3 (25 נקודות)

1. הוכח/הפרך: לכל $n \in N$ קיימת ערימת מינימום בעלת n איברים הממומשת ע"י עץ כמעט שלם, בה לכל איבר a בעומק $d(a)$ ואינדקס $r(a)$ מתקיים $\lfloor log_2(r(a)) \rfloor \leq d(a)$ תזכורת: אינדקס של איבר הוא מיקומו בסדרה הממוינת של n האיברים.
2. הוכח/הפרך: קיים מימוש לערימת מינימום כך שבכל רגע נתון מספר האיברים השונים זה מזה הוא מהטווח $[0, log\ n]$ (כאשר n הוא מספר האיברים הנוכחי בערימה), ופעולות $DeleteMin$ ו- $Insert$ יבוצעו בסיבוכיות זמן: $O(log\ log\ n)$ (כל שאר הפעולות יבוצעו בסיבוכיות שנלמדה עבור המימוש של ערימה כעץ כמעט שלם).

   שימו לב: בסעיף זה ייתכנו איברים בעלי אותו מפתח.
3. הוכח/הפרך: קיים מבנה נתונים התומך בפעולות הבאות:

   $Init(S)$ מקבלת מצביע לערימת מינימום S הממומשת באמצעות עץ בינארי כמעט שלם ומאתחלת את המבנה. סיבוכיות זמן: $O(1)$

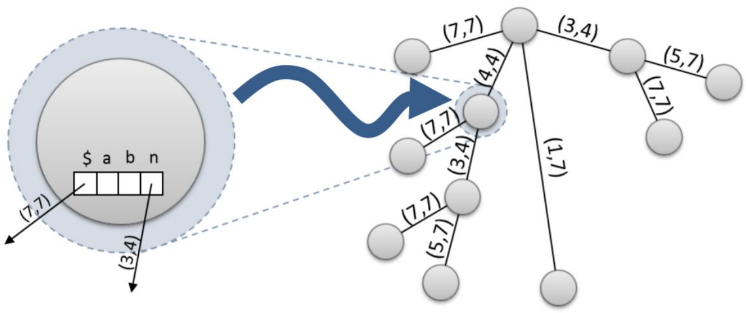
   $Select(k)$ מחזירה את האיבר ה- k בגודלו מבין איברי הערימה S

   סיבוכיות זמן: $O(h_k+1)$, כאשר h_k הוא גובה הצומת של האיבר ה- k בגודלו.

   תזכורת: הגובה של צומת בעץ הוא מספר הקשתות במסלול הפשוט הארוך ביותר מהצומת לעלה הנמצא בתת העץ שהצומת הוא שורשו.

שאלה 4 (20 נקודות)

בשאלה זו נעסוק בעץ סיומות דחוס מעל א"ב Σ בגודל קבוע. הניחו כי אינדקס תחילת מחרוזת הוא 1. נתון עץ סיומות דחוס T אשר נוצר ע"י אלגוריתם הקופסה השחורה. כפי שנלמד בקורס, כל צומת מחזיק מערך בגודל $|\Sigma|+1$ המייצג את הקשתות היוצאות ממנו (ראו דוגמא). בעקבות תקלה, אבדה המחרוזת אשר העץ T מייצג. בדוגמא להלן נאבדה המחרוזת banana מעל א"ב {a,b,n}.



מצאו אלגוריתם אשר בהינתן עץ סיומות דחוס T משחזר את המחרוזת s אליה שייך העץ T בזמן $O(|s|)$.

שאלה 5 (15 נקודות)

בשאלה זו נדון במחרוזות מעל א"ב קבוע סופי נתון Σ.

1. הציעו מימוש למבנה נתונים התומך בפעולות הבאות:

$Init()$ מאתחלת את מבנה הנתונים.

סיבוכיות זמן: $O(1)$.

$Push(c)$ מכניסה את התו $c \in \Sigma$ למחסנית.

סיבוכיות זמן: $O(1)$ משוערך ביחד עם $ClearStack$

$PopString()$ מכניסה את המחרוזת שנמצאת במחסנית למבנה, ולאחר מכן מרוקנת את המחסנית. המחרוזת מתחילה בתחתית המחסנית ונגמרת בראש המחסנית (המקום בו הוכנס התו האחרון).

המבנה מאפשר כפילויות – ייתכן שהמחרוזת שבמחסנית כבר קיימת במבנה.

סיבוכיות זמן: $O(1)$ משוערך ביחד עם $Push$

$GetCommon()$ החזר את המחרוזת הנפוצה ביותר שקיימת במבנה. אם קיימות מספר מחרוזות נפוצות ביותר, החזר את הקטנה ביותר לקסיקוגרפית.

סיבוכיות זמן: $O(|s|)$, כאשר s הוא אורך המחרוזת שתוחזר.

2. נסמן ב- s את המחרוזת האחרונה שהוכנסה למבנה בפעולת $PopString$.

הראו כיצד ניתן להרחיב את מבנה הנתונים מסעיף א', כך שיתמוך בפעולה הבאה, מבלי לפגוע בסיבוכיות של שאר פעולות המבנה:

$CountSubstring(r)$ מדפיסה את מספר הפעמים שהמחרוזת r מופיעה כתת-מחרוזת של s.

במידה ועוד לא הוכנסה אף מחרוזת למבנה, הפעולה תחזיר 0.

סיבוכיות זמן: $O(|r|)$.

בהצלחה!