

## אלגוריתם קרוסקל:

רעיון: שיטה חמדנית - בכל שלב בחר את הצלע עם המשקל הכי נמוך כל עוד זה לא סוגר מעגל. כדי "לבדוק" שצלע לא סוגרת מעגל באופן יעיל משתמשים במבנה נתונים הנקרא  $disjoint$  sets.

(בשמו האחר:  $union - find$ )

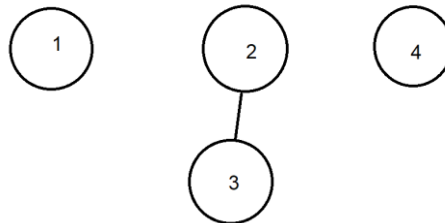
תיאור מבנה התונים:

אתחול - מערך של  $n$  קודקודים כאשר לכל אחד יש קבוצה משל עצמו והוא מנהיג הקבוצה.  
 $makeSets(4) [1,2,3,4]$  -



איחוד:

לוקח 2 קודקודים ומאחד בין הקבוצות שלהם (לאו דווקא ישירות ביניהם) אחד מהם יהיה מנהיג  
 $union(2,3) [1,2,2,4]$  - הקבוצה.



אם נאחד את 4 ל 3 אז 4 יהפוך להיות מחובר ל 2 מלמטה (ולא ל 3)

חיפוש: מציאת מנהיג הקבוצה של קודקוד:  
 $find(3)$  יחזיר 2.

שיפור 1: חיבור לפי דרגה: כאשר מאחדים 2 קבוצות, נעדיף לחבר את מנהיג הקבוצה הקטנה למנהיג הקבוצה הגדולה ולא להיפך כדי לא ליצור מסלולים ארוכים.  
 כי אז כל חיפוש יהיה: מנהיג קטנה - מנהיג גדולה - (מסלול ארוך) שאר חברי הקבוצה ולכן עדיף: מנהיג גדולה - מנהיג קטנה - (מסלול קצר) שאר חברי הקבוצה.  
 שיפור 2: כיווץ מסלולים: בזמן החיפוש של מנהיג הקבוצה, כאשר נמצא אותו, נעדכן את כל הקודקודים שהיו על מסלול החיפוש להתחבר ישירות למנהיג (ולא רק הקודקוד שחיפשנו עבורו).

אם לא משתמשים בשיפורים - סיבוכיות החיפוש והאיחוד יכולה להיות  $O(n)$ .  
 אם משתמשים בשיפור 1 - סיבוכיות יורדת ל  $O(\log(n))$   
 אם משתמשים גם בשיפור 2 - סיבוכיות ממוצעת יורדת ל  $O(\alpha(n))$  כאשר  $\alpha(n)$  - כמה פעמים צריך להפעיל  $\log$  על  $n$  כדי לחזור ל 1.  
 דוגמא:  $\alpha(2^{2^{100}}) = 4$ . כי:  $\log(\log(\log(\log(2^{2^{100}}))) \approx 1$ .

האלגוריתם: (של קרוסקל)

בהינתן גרף  $G$  נבנה מערך  $E$  של צלעות הגרף כאשר לכל צלע  $e$  נשמור את קצוות הצלע:  $e.v_1, e.v_2$  ואת המשקל שלה:  $e.w$ .  $n = |V|$ .

1. מיון את  $E$  מהמשקל הקטן לגדול.
2. אתחל מבנה  $UF$  בגודל  $n$  - נקרא לו  $U$ .
3. אתחל:  $T = \emptyset$ .
4. עבור כל  $e \in E$

a. בדוק האם:  $U.find(e.v_1) \neq U.find(e.v_2)$

i. אם כן (קבוצות שונות) אז:

1.  $U.union(e.v_1, e.v_2)$

2.  $T = T \cup \{e\}$

3. אם  $|T| = n - 1$  החזר את  $T$ .

5. החזר שאין עץ פורש.

סיבוכיות:  $O(|E| \cdot \log(|E|) + |V| + |E| \cdot \alpha(|V|))$

מיון מערך הצלעות הוא:  $O(|E| \cdot \log(|E|))$

אתחול מבנה UF הוא:  $O(|V|)$  בניית המערך.

לולאה ראשית: עוברים על כל צלע ועבור כל אחת מחפשים את קבוצות 2 קצוות הצלע באמצעות מבנה הנתונים ומחברים אותם אם אפשר. אם משתמשים בכל השיפורים של המבנה אז סיבוכיות

חיפוש ואיחוד היא:  $O(\alpha(n))$

לכן סה"כ:  $O(|E| \cdot \alpha(|V|))$ .

$O(|E| \cdot \log(|E|) + |V|)$  סה"כ: