hanoch@ cs.tau.ac.il

O(h)   log(h)   ...

O(h)   O(n)   get min

O(h)   O(h log h)   find min

O(k·h)   O(h + k log h)   O h log h

O log(h)   O log(n)   delete

ht K log k

O(n)
log(h)

סיבוכיות של Dijkstra

Insert(x, Q) $n = |V|$

min(Q) n

del min(Q) n

Decrease key(x, Q, Δ) $m = |E| \leq n^2 - n$
         ↓
     אפשר ע״י

Delete
+ insert

find ... עבור delete ...
         ... מוצא ...

$n \log n + m \cdot \log n$
שיפור

... decrese
    key

... $\log n$ ...

$O(n \log n + m \cdot \log n)$

$m = O(n^2)$ ... $m \log n$ ...

עצי בינומיים

$B_0 = 0$

$B_1 =$

$B_i =$ $\quad B_{i-1} \quad B_{i-1}$

$B_2 =$

כל עלה בעומק שלו...
תיאור שלו B_i ... כל עלה...
... בגובה ... כל עלה בגובה.

B_{i-1}

$B_i =$

$B_{i-1} \qquad B_{i-2} \qquad B_2 \qquad B_1 \qquad B_0$

מספר הצמתים ... אשר בעלה ...

$B_i = $ הגובה ... מספר $\qquad$ של

כ"א

$B_{i+1} =$ $= \quad B_i \quad B_{i-1} \quad B_1 \quad B_0$

B_i

מספר הצמתים
בכל שכבה)

גובה זול יותר.

$$|B_{k+1}| = \overset{B_k}{\triangleleft}\,\overset{B_k}{\triangle}\, = 2|B_k|$$

קומבינטורי $|B_k| = 2^k$

$depth(B_k) = k$

$depth\ B_{k+1} = depth(B_k) + 1$ בנוסח האינדוקטיבי

כאשר אנו כבר ברקורסיה של (B_k), גובה אומרת שלו גדל נשאר 0.

כאשר אנו מגדילים על ידי גורם של n אחד הגובה h לוגריתמי

$$\log n$$ גובה

ערימה בינומית

ערימה בינומית הינ סופל סט של עצים בינומית כאשר כל אחד מהם מקיים את שני תנאים:

$\dagger$ סידור נכון - heap order העצים מקיימים

כאשר הצמתים מחוברים אחד אל השני ב doubly linked list


```
     (3)          (7)      (5)
    /   \          |
  (9)   (8)       (6)
   |
  (22)
```

min מצביע על הצומת המינימלי של הערימה הבינומית

איחוד ערימה - מיזוג מערכות הבינומית ל ערימה אחת.

Linking סדר אותם מלמעלה למטה לפי סדר

השוואה מחברים לפי גודל מפתח הבורר

Melding

B_5 B_4 B_2 B_1

h_1 B_4 B_3 B_1 B_0

h_2 B_4 B_3 B_0 + $\rightarrow$ לכל

_____

B_5 B_4 X B_2 X X   Link

לכל שורה מספר סדר

חוזר ברגע, יכול/ני

לעצם (שלנו) יוצגת מספר על העצים הבינומיים שיש ל'

$\forall\ 0 \le k \le \ldots$ $2^k = |B_k|$ י' לכל ה' לכל

לכל אחת מהפעולות יתקיים

find min   כי יהיה ברשימה $O(9)$

Insert x

מסירת אל B_0 ואחרי Meld

$O(\log)$

del min

composition עיצורים, ואחרי - עוצר קורבבתית

$\leftrightarrow$ לכל שורה   ברגעי, ואחרי Meld $O(\log 10)$

רשימות אחרות יכולת על $O(\log)$ לעצם שלנו
אחת אשר עצם
לאחרות

Dijkstra   כי העצם הקטן ללא X אחרי העצם

אוחזר X, ראשי עד עצם ילד ל X, לעצם
כולו

נשמור

לאחר כל עצם X פעם delete ואחרי נשמור יוצא

$O(\log)$

decrease key (x, h, A) מזהה θ אותו מוצאים

אנחנו מורידים בזה לשורש ודואגים שיהיה מינימלי $\log(A)$

$O(\log(A))$ סיבוכיות כוללת

$O(\log(A))$

000
001
010
011
100
101
$\vdots$

$O(h)$

Incerement של מונה בינארי

$\vdots$ "1"

צריך 1!

105 ... 0000000 ... $0 -$

$N = 0$ 1 5 $0 -$

$h+h = 2h + h$

$O(\log n)$

Amortized $O(1)$

$O(h)$

Φ $O(n)$

Lazy melds

סעיף אחר ראו כאן שיא אותו מניימ קום מניימ אחר פינה של

כזה מניימ, מחבר אליו את כל המניימ האחרים -

כזה לעשות delete min בפעם אחרת

del-min

טענה אבחנה min-del כאשר אין לחפש : כולי נעשינו

ואלו עם רשימת של כולם האחרים, אם (3)log היא אין,

לכן זה לעשות

ולכן אפשר איטרלציות של כל זה ככה

בסוג כל מסלול של זה עץ כי כל במקום,

זה פיני כי מין ואו אוליל אשר ויין יותר
carry

ולכן כולי של איטרצ... של כל רשימות/

ולמצוא על כל גודל logn זה פעם אולי/

ולכן.

? delete של 5''(...)אחר ו אם

min

(פוטנציאל) (...) עצים = $\Phi(c)$ = אם האיטר.

ברורה של עצים על אחת 2. א. logn

כי של delete min כי אם רמז דרך δ על כל עצים (...)

שורש אולי פעם

| delete min | ככלל אולי (1)c אם רמז כי min |

כאן מכולם, כי ככלי כאשר כולם כאן עם

(...) כי וליל רמז על רמז ה ואלי/...

(...)

Amotized $= (\text{\# links} + \text{max-rank}) - (\text{\#links} - \text{max} - \text{rank})$

כמה ל link

Array setup — every link removes tree

כולם del min יותר logn כמכה נעשה

Find min $O(1)$ w.c.

Meld — $O(1)$ w.c. ⟶

Insert — $O(1)$ w.c. ⟶

<div dir="rtl">

בגלל שאנחנו רוצים זה לכל כזה נכתוב סימון, אבל כשיש לנו מעלה

ובמקרה הכי גרוע ככה סכום, ואנחנו מחשבים

$\overline{\Phi}(v) = m$ סכום מעלות   כמו שמגדירים

</div>

delete min   Amortized $O(\log n)$

<div dir="rtl">
נמחק בגרף/ליסט
</div>

Decrease key $decrease(v)$ את סדר רוצה לשנות

A Amortized $O(1)$ נרצה

<div dir="rtl">
cost של amortized כי $ci=O(1)$ רוצה   m פעולות
</div>

{ sum קושי? }

<div dir="rtl">

Decrease key נסמן

את אחרת שאנחנו רוצים מקצר ⟶ X   אבא

קושי ⟶ בגלל

אם היינו X על הצומת אחרת

</div>

$O(1)$ ~~זמן שכל פעם~~

<div dir="rtl">

אנחנו ברצף בגלל כל מיני.

אבל זה במקרה גרוע יכול לקרות,

אז c_i יכול אחר רוצה log הזמן.

</div>

אבל עכשיו בחירה אחרת עצמנו מעל לוג של העץ (כי העץ עצמו)

$\log n$

Rank of the tree = degree of the root

כאשר יש עץ – דרגת הדרגה של

ברוב המקרים Succesive linking – Delete min

של כל אחד (מהעץ).

בכל עלה new meld Insert

Cascading Cuts fr decrease key

Node
שולחים עץ ... שיש ... של

אבל עץ ... – 2 של העץ

כאשר אחרי ... שלנו ... ... הכל כאשר

של (O(n) ... ... של שלנו.

עכשיו הבעיה – ... ... של המצב

של הבעיה ... ... של

deletemin → $\log n$ ... של

... ... של שלנו.

בחירות $O(1)$: כל שורה ... ... של שלנו,

אבל Node ... ... decrease ... של אותה
min
↓

פעם ... ... של

decrease key

פעם $O(1)$ של ... של
(... של
... של) $O(\log n)$ ... של)

cascading cuts ... delete –
del min ... של

אם נעשה $O(\eta)$ ...

... decrease a key

... "...0%..."

...

... $O(\log n)$ ...

decreasekey -- deletemin ...

... Insert ...

decrease N ... key

... $\log n$.