הרצאה 2

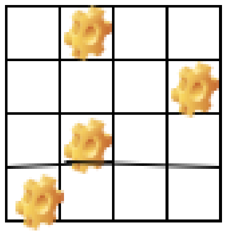
איתי וייסמן

5 במאי 2019

1 אלגוריתם DFS

דוגמה: רובוט סורק אזור מסויים במטרה לגלות מוקשים.

- נייצר את אזור הסריקה על ידי סריג (מטריצה)
- הרובוט יכול להתרדם בכל צעד ימינה, שמאלה, למעלה ולמטה.



לדוגמה:

גישה טבעית לסריקת האיזור היא התקדמות עד גילוי מוקש, סימון המשבצת וחזרה אחורה עד למשבצת שממנה ניתן להתקדם.

זו גישה שנקראת "חיפוש לעומק".

1.1 אלגוריתם חיפוש לעומק (Depth-First-Search) - DFS

מנסה להתקדם כמה שיותר לעומק בגרף.

כאשר נבקר בצומת $v \in V$ אם יש קשת $(v,u) \in E$ אל צומת u שעוד "לא התגלה" נחצה את הקשת ונמשיך את הסיור מהצומת u.

<u>מטרה:</u> יש לגלות את כל הצמתים בגרף.

פורמאלית:

<u>סימונים:</u> נסמן ב$d[v]$ את זמן הגילוי של v וכן ב-$\Pi[v]$ את הצומת שגרם ל-v "להתגלות".

Algorithm 1. Input: a directed graph $G=(V,E)$, a starting vertex $s \in V$

Output: for all $v \in V$, it's time of discovery

(1) for all $v \in V$ **do**

1.1 $d[v] \leftarrow 0$, $\pi[v] \leftarrow null$

1.2 mark all edges "unused".$i \leftarrow 0, v \leftarrow s$

.1.3 $i \leftarrow i+1, d[v] \leftarrow i$

.3 while there are unused out-edges from v **do**:

3.1 choose unused edge (v,u) mark it as "used".

3.2 if $d[u]=0$ then **do**:

3.2.1 $\pi[u] \leftarrow v, v \leftarrow u, i \leftarrow i+1, d[v] \leftarrow i$

.4 if $\pi[v] \neq null$ then **do**

4.1 $v \leftarrow \pi[v]$

4.2 goto (3)
.**5 else if** there is $u \in V$ with $d[u] = 0$ then **do**:
5.1 $v \leftarrow u$
5.2 goto (2)
6 stop

1.2 נשים לב:

1. בהרצות שונות של האלגוריתם נוכל לקבל פלטים שונים אבל בכולם נקבל "יער" שבו כל צומת מופיע באיזשהו עץ מכוון.
2. האלגוריתם לא בהכרח מוצא מרחקים קצרים

2 אלגוריתם DFS בצורה רקורסיבית

סימונים:
נסמן ב-$color[u]$ - צבע של צומת u מתוך הקבוצה $\{white, gray\}$
נסמן ב-$f[u]$ - זמן הנסיגה מצומת u
$Adj[u]$ - אוסף השכנים של u.

Algorithm 2. **Input**: a directed graph $G = (V, E)$
Output: for all $v \in V$, it's time of discovery $d[u]$ and it's time of retreat $f[u]$.
$DFS(G = (V, E))$:
for each $u \in V$ **do**: $color[u] \leftarrow white, \pi[u] \leftarrow null$
do $i \leftarrow 0$
for each $u \in V$ **do**
if $color[u] = white$ then **do**:
$DFS - VISIT(u)$

אלגוריתם 3. $DFS - VISIT(u \in V)$:

Algorithm 4. $color[u] \leftarrow gray$
$i \leftarrow i + 1$
$d[u] \leftarrow i$
for each $v \in Adj(u)$ **do**
if $color[v] = white$ then **do**:
$\pi[v] \leftarrow u$
$DFS - VISIT(u)$
$i \leftarrow i + 1$
$f[u] \leftarrow i$

2.1 ניתוח זמן הריצה

זמן הריצה של לולאת האתחול $\Theta(|V|)$.
נסמן ב$T(|DFS - VISIT|)$ את מס' הפעולות המבוצעות בקריאה לפרוצדורה $DFS - VISIT$ עבור הצומת u. נשים לב כי $DFS - VISIT$ נקראת בדיוק פעם אחת עבור צומת u כאשר u צומת "לבן". מיד בכניסה לפרוצדורה, u נצבע ב"אפור".

בנוסף, מס' הפעולות בלולאת ה"for" של הפרצדורה הוא לינארי במס' השכנים של u ב-G.
לכן סיבוכיות הקריאות ל$DFS-VISIT$ הוא:

$$\sum_{u\in V} T(DFS-VISIT(u)) = \sum_{u\in V} \Theta(|Adj(u)|) = \Theta(|E|)$$

ולכן סך זמן ריצה האלגוריתם הוא $\Theta(|V|+|E|)$.

2.2 תכונות האלגוריתם

2.2.1 תכונה בסיסית

בסיום הריצה נקבל יער של עצי DFS אשר המבנה שלהם משקף את הקריאות הרקורסיביות ל-$DFS-VISIT$.

2.2.2 צבעים

v הוא צאצא של u בעץ DFS אםv התגלה כאשר u היה אפור (ולפני שקבענו ערך ל-$f[u]$).

2.2.3 תכונת הסוגריים

משפט 1: משפט הסוגריים. בכל חיפוש לעומק של גרף מכוון או לא מכוון, G, לכל שני צמתים $u,v\in V$ מתקיים אחד משלושת התנאים הבאים:

1. האינטרוולים $(d[u],f[u])$ ו-$(d[v],f[v])$ זרים לחלוטין ואין קשר של אב קדמון-צאצא בין הצמתים.
2. האינטרוול $(d[u],f[u])$ מוכל ממש באינטרוול $(d[v],f[v])$ וu צאצא של v בעץ DFS.
3. האינטרוול $(d[v],f[v])$ מוכל ממש באינטרוול $(d[u],f[u])$ ו-v צאצא של u בעץ DFS.

הוכחה:

נבחין בין שני מקרים:

1. במקרה בו $d[u]<d[v]$, נבחין בין שני תתי מקרים.
   (א) $d[v]<f[u]$ - אזי v התגלה כאשר u היה אפור ולפני הנסיגה מ-u. ולכן v הוא צאצא של u, ולכן $f[v]<f[u]$ ונקבל כי $(d[v],f[v])\subset(d[u],f[v])$.
   (ב) $d[v]>f[u]$ - אזי נקבל כי מאופן התקדמות האלגוריתם, יתקיים $d[u]<f[u]<d[v]<f[v]$ ולכן האינטרוולים זרים לחלוטין. היות וv התגלה אחרי הנסיגה מ-u אין בינהם קשר של אב קדמון-צאצא.
2. $d[v]<d[u]$, באופן דומה מבחינים בין שני תתי מקרים באופן סימטרי למקרה 1.

■

2.2.4 תכונה של צאצאים ביער DFS

משפט 2: (המסלול הלבן)
ביער DFS של גרף G (מכוון\לא מכוון), צומת v הוא צאצא של צומת u אם ורק אם בזמן $d[u]$ הזמן בו u התגלה, ניתן להגיע ממנו ל-v על מסלול שמורכב כולו מצמתים לבנים.
הוכחה:

1. כיוון ראשון - נניח שv צאצא של u, יהיה w צומת על המסלול בין u ל-v בעץ DFS כך ש-w צאצא של $\underline{u}$. לכן ממשפט 1, אנו למדים כי $d[u]<d[w]$ ולכן w היה לבן בזמן $d[u]$.

2. כיוון הפוך - נניח בשלילה שיש מסלול לבן מu ל-v בזמן $d[u]$ אבל v לא נהיה צאצא של u בעץ DFS ונניח ש-v הוא הצומת הראשון על המסלול הלבן שאינו הופך לצאצא של u. יהיה w הצומת שלפני v על המסלול הלבן כך ש-w הופך לצאצא של u (יתכן כי $w = u$). ממשפט 1 אנו למדים כי מתקיים $f[w] \leq f[u]$. נשים לב כי v חייב להתגלות אחרי u אבל לפני הנסיגה מ-w, כלומר $d[u] < d[v] < f[w] \leq f[u]$. ממשפט 1, נקבל כי $(d[v], f[v]) \subset (d[u], f[u])$ ולכן v צאצא של u בעץ DFS. $\lightning$ סתירה.

■