מ.מ.למדמ"ח ~ סיכום 22.5.2023

שחר פרץ

22 במאי 2024

1 קצת על איטרטורים

גנרטור: אויבטקט שמייצר איברים. בפייתון, ממומש ע"י פונקציה. הפונקציה מממשת את זה באמצעות הרצת קוד בעבור חישוב כל ערך שהיא מחזירה. בפייתון: נשתמש ב־yield. פונקציית גנרטור מחזירה גנרטור.

בעת השמה ללואת for אובייקט משוג כלשהו, הוא קורא לפונקציה iter שנבקשת מהאבוייקט להחזיר איטרטור. אם היינו שמים בו גנרטור, אז היה מוחזר גנרטור. אם היינו שמים רשימה, הוא היה מחזיר "איטרטור של הרשימה" – איזשהו אירטור שמשתמש ברשימה, ומאחסן רק את האינדקס בתוכו. הלולאה תעצור כאשר היא תקבל StopIteration. היא למעשה while עד שהאיטרטור נגמר (באופן שקול).

למעשה:

```python
for x in [1, 2, 3]: print (x)
  i = iter([1, 2, 3])
# is like
while True:
  try:
    x = next(i)
  exepct StopIteration:
    break
print(x)

```

הפונקציה iter קוראת למתודה __iter__. כדאי לציין שהחל מ־python 3 יתקיים ש־dict שומר על סדר ההכנסה. הערה: אפשר לעשות:

```python
for x in L := [1, 2, 3]:
  print(x)
  L.pop()
```

וזה ישנה את ערך x בהתאם. **מוסכן לשנות את האובייקט שחוזרים עליו על ה-runtime** כי זה רעיון רע.

עבור dict, לדוגמה:

```python
D = {1: 1, 2: 2}
for x in D:
  print(x)
  D.pop()
```

אז יקפוץ RunTimeError.

סה"כ, גם אם חייבים לשנות אובייקטים בזמן חזרה, תיהיו בטוחים מה אתם עושים ותיזהרו מאוד.

2 דקדוק (Grammar)

בעת הרצת קוד פייתון, האינטרפטר מפרש את הקוד שכתוב. הדבר הראשון שנעשה הוא בדיקת הסינטקס, שהוגדר בעבור השפה. יש דקדוק פורמלי של השפה. הדקדוק מגדיר את השפה, ומגדיר את החוקים עבורם יתפרשו המשפטים. הדקדוק מגדיר את השמפטים (תוכניות) החוקיים בשפה מבינה מבנית.

בהינתן קלט, האינטרפטר צריך להחליט/להבין האם הקוד תקין מבנית.

בגדול, דקדוק מוגדר ע"י:

1. משתנים (Variables): V
2. א"ב (alphabet/Terminals) : Σ
3. כללי היסק/גזירה (Rules): R
4. מתשני התחלה (start-variables): S

דוג':

$$\begin{array}{c} G_1\colon (V,\Sigma,R,S) \\ V\colon \{A,B\} \\ \Sigma\colon \{0,1,\#\} \\ R_1: \begin{array}{ccc} A & \to & 0A1 \\ A & \to & B \\ B & \to & \# \end{array} \\ S\colon A \end{array}$$

איך מייצרים מילה מתוך דקדוק?
מתחילים במשתנה ההתחלה.
כל עוד יש בידנו משתנים:

- נבחר משתנה שישש לנו
- נבח רכלל גזירה עבורו
- נפעיל את כלל הגזירה

לדוגמה: $S \to A = 0A1 = 0B1 = 0\#1$

נסמן ב־$L(G)$ את אוסף המילים החוקיות ע"פ הדקדוק. נסמן מילה חוקית W ע"י $w \in L(G)$. בדוגמה הזו, ניתן להגיד $''0\#1'' \in L(G)$.
$L(G)$: השפה שמוגדרת ע"י הדקדוק G

במקרה הזה, $L(G_1) = \{0^n\#1^n \mid n \in \mathbb{N}\}$.

$$G_2\colon (V,\Sigma,R,S) \tag{1}$$

$$V = \{S\} \tag{2}$$

$$\Sigma = \{a,b\} \tag{3}$$

$$R\colon \begin{array}{ccc} S & \to & aSb \\ S & \to & SS \\ S & \to & \epsilon \end{array} \tag{4}$$

כאשר ϵ היא המחרוזת הריקה. דוגמאות:

$$ab \in L(G_2),\ aaabbb \in L(G_2),\ abab \in L(G_2)\ \epsilon \in L(G_2)$$

ובכלליות: אם נגדיר $a = (, b =)$ נקבל את מספר הקומבינציות החוקיות של סוגריים.

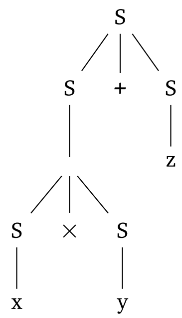
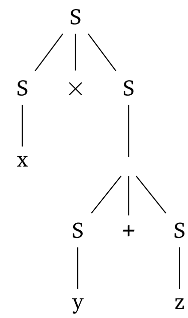
דוגמה אחרונה:

$$\begin{array}{l} G_3\colon (V,\Sigma,R,S) \\ V = \{S\} \\ \Sigma = \{x,y,z,\times,+\} \\ R\colon \begin{array}{ccc} S & \to & S * S \\ S & \to & S + S \\ S & \to & \times \\ S & \to & y \\ S & \to & z \end{array} \end{array}$$

לדוגמה:

$$x \times y + z \in L(G_3),\ x + y \in L(G_3), x + x + x + x \in L(G_3)$$

הוכחה שהדוגמה הראשונה תקינה:

כלומר השפה אמביציונית (ambigous); ניתן לגזור מילה במס' דכי גזירה שונים.

כל הדקדוקים שראינו הם ממשפחה הקרוייה "דקדוקים חסרי הקשר" – grammer free contect – CFG.

דקדוקים אלו מטילים מגבלה כלשהי: בכל כלל גזירה, בצד שמאל יופיע משתנה יחיד, שמופיע פעם אחת. בדקדוקים אלו אין צורך להתבונן במה קדם לתו שאנו נמצאים בו. דוגמא לדקדוק שאינו כזה: $aS \to ab, Sb \to aa$.

דקדוק חסר הקשר כל כללי הגזירה נראים $A \to \ldots$.

2.1 שאלות

נגיד ש־G_1, G_2 דקדוקים חסרי הקשר.

- האם $L(G_1) = L(G_2)$? $\iff$ האם הדקדוקים מגדירים את אותה השפה?
- האם $L(G_1) \subseteq L(G_2)$? $\iff$ האם השפה המוגדרת ע"י G_2 מכילה את השפה שמוגדרת ע"י G_1?
- האם $L(G_1) \cap L(G_2) \neq \emptyset$? $\iff$ האם יש חיתוך בין שתי השפות?
- האם קיימת מילה בשפה עבורה יש שני עצי גזירה שונים? $\iff$ האם הדקדוק הוא אמביציוני?

4 השאלות האלו לא ניתנות לחישוב. פרט לגודל האינסופי של $L(G)$, אי אפשר לפתח אלגוריתם המבצע השוואה בין החוקים, בכל המקרים.

שאלות שכן אפשר לענות עליהן:

- בהינתן דקדוק חסר הקשר G ומילה w, האם $w \in L(G)$?
- בהינתן דקדוק חסר הקשר G ומילה $w \in L(G)$, למצוא עץ גזירה כלשהו של w מ־G.

את השאלות האלו אפשר לפתור באופן יעיל.

רעיון: ננסה לעבור על כל הגזירות האפשריות.
בעיה: כאשר $w \notin L(G)$, התהליך יכול לא לעצור. אין כלל עצירה טוב.

פתרון: נמיר את הדקדוק צורה שנקראת Chomsky-Normal-Form – CNF, שבה קל יותר לענות על השאלות האלה. כל דקדוק חסר הקשר ניתן לכתיבה בצורת CNF.

2.2 CNF

בצורת CNF מותרים כללי גזירה מהצורה הבאה:

$$\begin{array}{ll} A \to A & A \in C, A \in \Sigma, \\ A \to BC & A, B, C \in V \\ S \to \epsilon & \end{array}$$

איך נמיר דקדוק לכזה?

$$\begin{array}{l} A \to BCD \\ A \to BA^*, A^* \to CD \end{array}$$

$$\begin{array}{l} A \to \epsilon \\ A \to S, S \to \epsilon \end{array}$$

$$\begin{array}{l} A \to aA \\ A \to BA, B \to a \end{array}$$

למעשה, כלל גזירה של מילה, הוא בהכרח עץ בינארי.