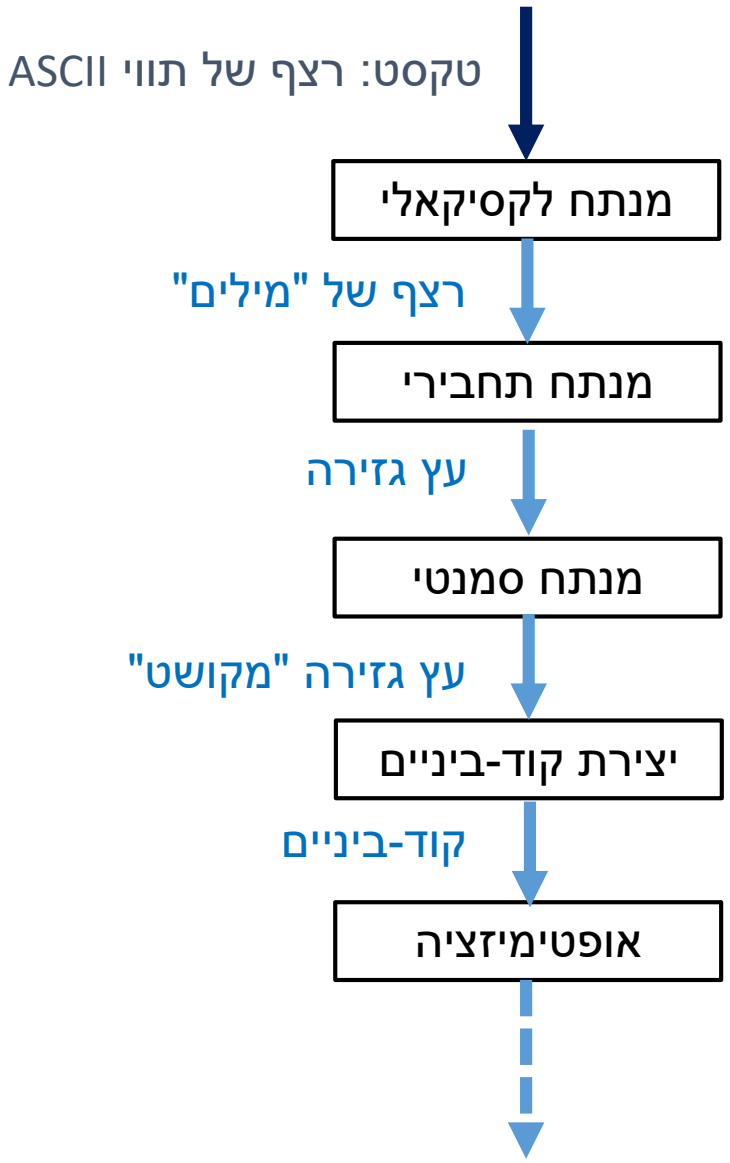
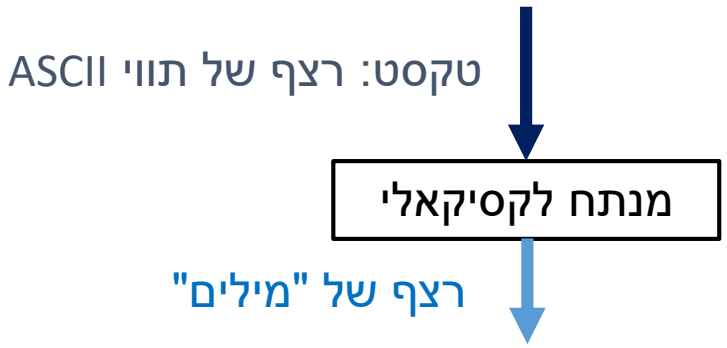
המנתח הלקסיקאלי


טקסט: רצף של תווי ASCII
מנתח לקסיקאלי
רצף של "מילים"
מנתח תחבירי
עץ גזירה
מנתח סמנטי
עץ גזירה "מקושט"
יצירת קוד-ביניים
קוד-ביניים
אופטימיזציה


טקסט: רצף של תווי ASCII
מנתח לקסיקאלי
רצף של "מילים"

תפקידו של המנתח הלקסיקאלי

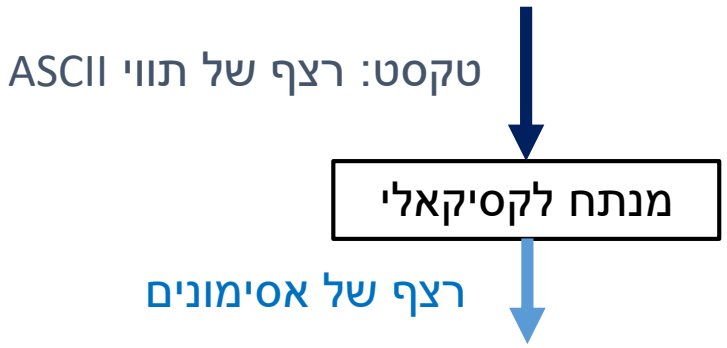
- זיהוי של "מילים חוקיות" : אסימונים
- מעקב אחרי המיקום בקלט
- דילוג על "תווים לבנים", הערות וכד'

אסימון Token

- המחרוזת שנקראה מהקלט:

  לֶקסֶמָה - lexeme

- סוג (בשקף הבא)
- מיקום בקלט (שורה, מספר התו בתוך השורה וכו')


טקסט: רצף של תווי ASCII
מנתח לקסיקאלי
רצף של אסימונים

סוגי אסימונים / סוגי מילים חוקיות

- מילים שמורות keyword . : if, for, int, main, ...
- מזהה, id. שם של אובייקט. : x, temp, int12, ...
- סימני הפרדה. : , , ; , (,) ,.....
- סימני אופרציה, אופרטורים. : + , - , && , ++ , =, ...
- מספרים. : 0, -3.56, 0.0, ...
- ...

דוגמא

if]+++<53.00for12)(1.5tem24$$$

אתגרים

- איך מגדירים חוקיות של כל סוג אסימון?
- איך מזהים אסימונים?
- מה עושים במקרה של קונפליקט בסיווג אסימון?

דוגמא

if]+++<for12)(15.tem24$$$

סוגי אסימונים / סוגי מילים חוקיות

- מילים שמורות keyword . : קבוצה סופית
- מזהה, id. שם של אובייקט. : קבוצה אינסופית
- סימני הפרדה. : קבוצה סופית
- סימני אופרציה, אופרטורים. : קבוצה סופית
- מספרים. : קבוצה אינסופית

ביטויים רגולריים (ב"ר)

ב"ר	השפה שהוא מתאר
ε הוא ב"ר	$[\varepsilon] = \{\varepsilon\}$
לכל $\sigma \in \Sigma$, σ הוא ב"ר	$[\sigma] = \{\sigma\}$
אם r_1,r_2 הם ב"ר אז $r_1 + r_2$ הוא ב"ר	$[r_1 + r_2] = [r_1] \cup [r_2]$
אם r_1,r_2 הם ב"ר אז $r_1 \cdot r_2$ הוא ב"ר	$[r_1 \cdot r_2] = [r_1] \cdot [r_2]$
אם r הוא ב"ר אז r^* הוא ב"ר	$[r^*] = [r]^*$
קדימות אופרטורים: (מהגבוה לנמוך) (1) * (2) • (3) +	

קשיים בכתיבת ביטויים רגולריים

$(i \cdot f + w \cdot h \cdot i \cdot l \cdot e + \ \ldots\ldots\ldots) \rightarrow (if + while + \ \ldots)$

(0+1+2+3+4+5+6+7+8+9) (0+1+2+3+4+5+6+7+8+9)*

➔(0+1+2+3+4+5+6+7+8+9)+

WHAT DOES “+” STAND FOR???!!!

ביטויים רגולריים מורחבים

- בחירה: מסומנת ע"י תו |
- איטרציה: אפס או יותר: *
- איטרציה חיובית: אחד או יותר: +
- שרשור: ניתן לוותר על סימן ה"נקודה"
- גרשיים: מבטלים משמעות ייחודית של תווים. למשל:

("/" | "*" | "-" | "+")

ביטויים רגולריים מורחבים

- קבוצת תווים: במקום שימוש בסימן הבחירה, ניתן לכתוב תווים באופן מפורש בתוך סוגריים מלבניות. למשל:

VOWEL [aeiou]

- טווח: במקום לרשום תווים באופן מפורש, אפשר לקבוע טווח של תווים (לפי הסדר בטבלת ascii). למשל:

DIGIT [0-9]

ALPHA [a-zA-Z]

ביטויים רגולריים מורחבים

- אופציונאלי: אפס או אחד מופעים של הביטוי. מסומן ע"י התו ?. למשל:

  (“-” | “+”)? [0-9]+

- אופרטור נקודה: כל תו מלבד ‘\n’ (newline).

חמדניות וקונפליקטים

- המנתח הלקסיקאלי פועל בצורה חמדנית:
  - מנסה למצוא את ההתאמה הכי ארוכה.
- במקרה של קונפליקט – מעדיף לבצע התאמה לפי כלל מוקדם יותר.

KW (if | for | int)

ID [a-zA-Z][a-zA-Z0-9]*

$$if12 \ \ is\ ID$$
$$if\ \ is\ KW$$

דוגמא הנחה: מותרים אפסים מובילים

- FIXED_POINT_NUM

(“+” | “-”)? [0-9]+ ”.” [0-9]+

- FLOATING_POINT_NUM

(“+” | “-”)? [0-9]+ ”.” [0-9]+ (e (“+”|“-”)? [0-9]+)

- REAL_NUM

(“+” | “-”)? [0-9]+ ”.” [0-9]+ (e (“+”|“-”)? [0-9]+)?