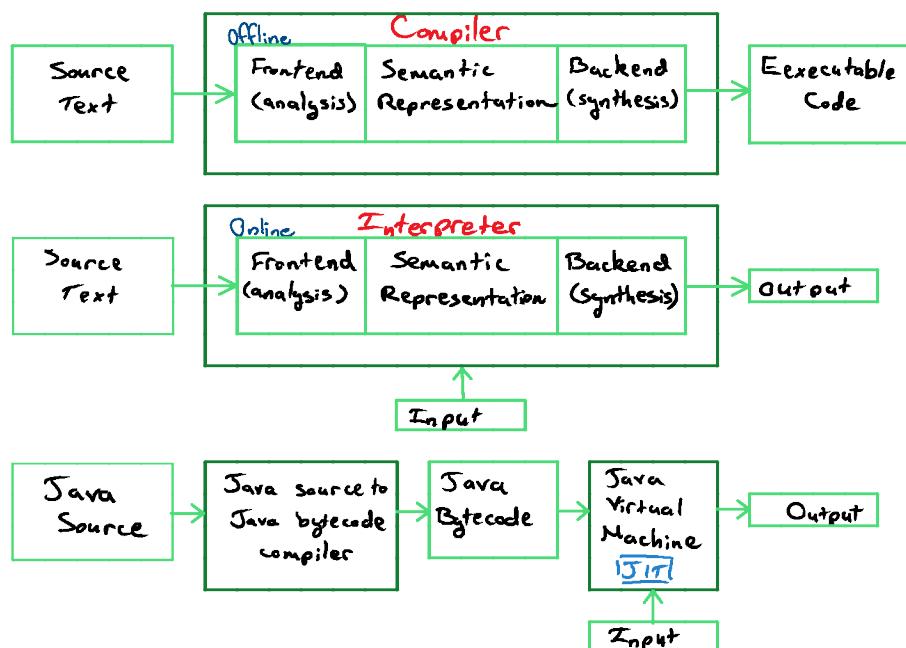


## למה קומPILE?

- \* קומPILE מטרת שילובם קווים נידולם לוחות קומPILEם נידולם.
- \* קומPILE שילובם קומPILEר קומPILEר שילובם.
- \* מטרת קומPILEר קומPILEר קומPILEר קומPILEר.
- \* קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר.
- \* קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר.
- \* קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר.
- \* קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר קומPILEר.

JIT Compiler ו Javascript - מהו JIT?



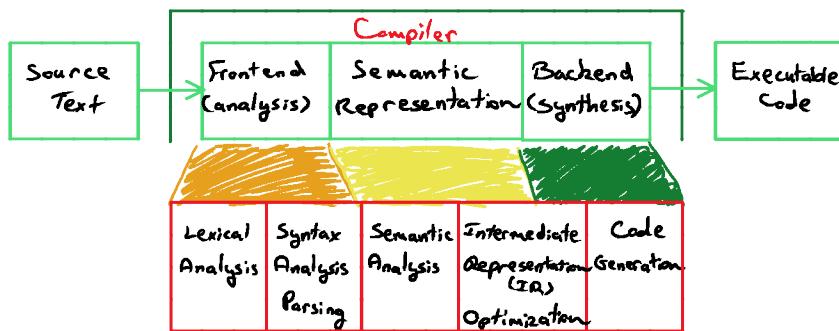
## מה זה JIT Compiler?

שיטה אוטומטית - נירער קומPILEר שילובם שילובם שילובם שילובם שילובם שילובם שילובם שילובם.

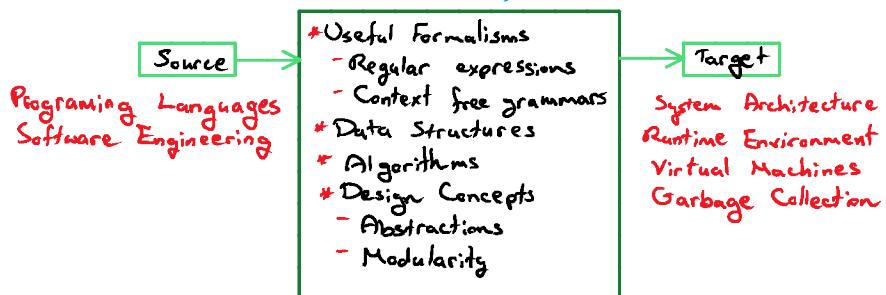
אנו.

- \* JIT קומPILEר נידולם ל- $L+M$  מילון  $M$  נידולם: JIT מילון נידולם.
- \* JIT קומPILEר נידולם שילובם JIT מילון נידולם.
- \* JIT קומPILEר נידולם שילובם JIT מילון נידולם.

## פונקציית סינון



? פונקציית סינון



\* מדריך חם טרי

\* מילון מילים טרי

\* הרפתק גולדן טרי

## טב נחים באלגוריתם

Tokens - טרמינלים - Lexical Analysis \*

Syntax Tree - עץ סינטטי - Syntax Analysis \*

- עץ סינטטי אובייקטיבי (AST) - Abstract Syntax Tree \*

Annotated Synt. Tree - עץ סינטטי מותג - Semantic Analysis \*

- IR - Intermediat Rep. \*

. IR -> Annotated Synt. Tree -> סימולטור -> מילון טרי \*

מילון טרי - מילון טרי טרי טרי טרי טרי טרי טרי טרי טרי \*

טב נחים באלגוריתם טרי \*

## שלבן מילויים

מזהה ופיזור טקסט. מילויים וטוקנים.

:Lexical Analysis \*

:Syntax Analysis \*

:Semantic Analysis \*

:Runtime \*

השאלה היא מהו מילויים וטוקנים?

## שלבן מילויים בפועל

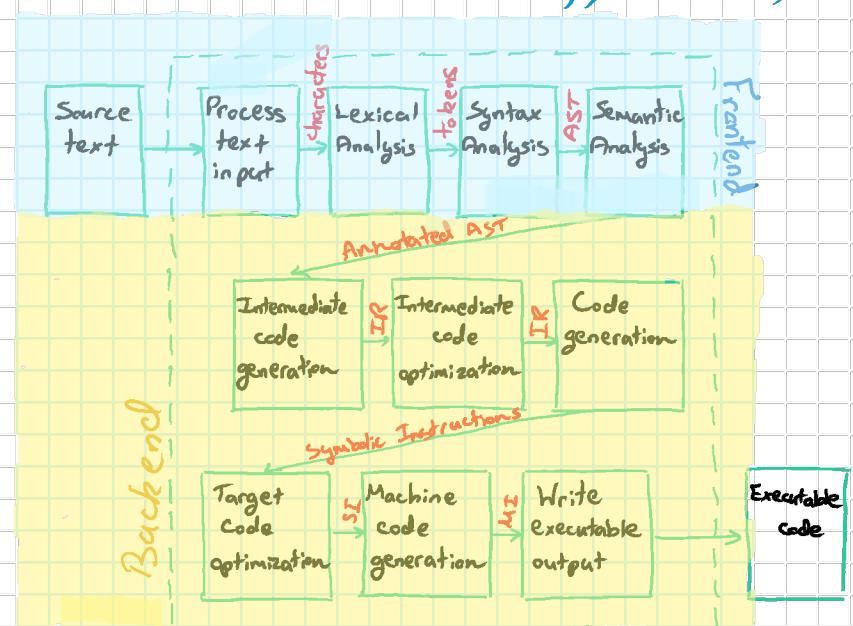
השאלה היא מהו מילויים וטוקנים?

טוקן הוא סדרת של סימני טקסט.

טוקן מוגדר (טוקן טרמינלי).

טוקן מוגדר (טוקן לא-טרמינלי).

## טוקן טרמינלי וטוקן לא-טרמינלי



## 3. סינטקיס

- \* הינה פונקציית סינטקיס שפונקציית פונקציית סינטקיס, ופונקציית פונקציית סינטקיס.
- מנגנון של פונקציית סינטקיס.
- מנגנון של פונקציית סינטקיס.

`for (int i=0; i<100; ++i) int t = x+y; : Loop Hoisting  
array[i] = x+y; => for (int i=0; i<100; ++i)  
array[i] = t;`

`for (int i=0; i<100; ++i) => for (int i=0; i<100; i+=5): Loop Unrolling  
delete array[i];  
delete array[i+1];  
delete array[i+2];  
delete array[i+3];  
delete array[i+4];`

- \* בודק שמיון מוגדר בפונקציית סינטקיס.
- . מנגנון של פונקציית סינטקיס.

`a=5  
b=a+3  
sum=0  
for (i in 1..100): sum+=i`      `a=3  
b=8  
sum=5050` : Constant Folding

## 4. פולימורפיזם

- \* פולימורפיזם.

(בבוקס) פולימורפיזם (בבוקס).

פונקציית פולימורפיזם.

## 5. פולימורפיזם

פונקציית פולימורפיזם.

פונקציית פולימורפיזם.

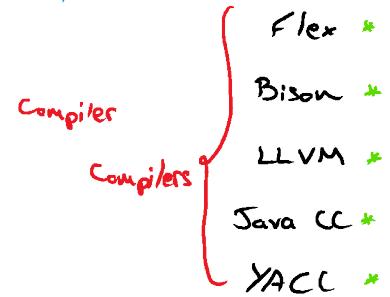
פונקציית פולימורפיזם.

פונקציית פולימורפיזם.

פונקציית פולימורפיזם.

פונקציית פולימורפיזם.

## הנתקים מ-Compiler



## PLD

\* קומpile = מtraducere מה-איסר ל-ויז'ן

(ללא פונקציית גורם גורם) (ללא פונקציית גורם גורם)

\* קומPILEר ימ' לאקסטר מילוקט וארצ'ט נ:

- צבאיים כבאים מדרישת אלגוריתם.

- צבאיים כבאים מדרישת אלגוריתם.

.(lexers/parsers: תרגום סימני כבאים כבאים)

\* קומPILEר נוירט לאקסטר נוירט

- לזרום כבאים כבאים

- מיל' frontend/backend

# ניתוח לקסיקלי של קוד המקור

10:58 AM Sunday, March 22, 2020

## לטוטו:

- \* מילון גיבובים נוצרו מהלך:  
 $x = b * b - 4 + a * c$
- \* מילון גיבובים נוצרו מהלך:  
 $\langle id, x \rangle \langle eq \rangle \dots \langle id, c \rangle \langle kind, values \rangle \in \text{tokens} : (\text{Token})$
- \* מילון גיבובים נוצרו מהלך:  
 $\text{whitespace}, \text{symbol}, \text{operator}$  ועוד.
- \* מילון גיבובים נוצרו מהלך:  
 $\langle pattern, attributes \rangle \in \text{patterns} : (\text{Pattern})$

לטוטו – מילון גיבובים נוצרו מהלך:  
 $\langle pattern, attributes \rangle \in \text{patterns} : (\text{Pattern})$   
 $\langle id, x \rangle \langle eq \rangle \dots \langle id, c \rangle \langle kind, values \rangle \in \text{tokens} : (\text{Token})$   
 $\langle pattern, attributes \rangle \in \text{patterns} : (\text{Pattern})$

## טבלת תיבות:

|             |  |
|-------------|--|
| $x$         | הויר חיבור 'x' מילון גיבובים                   |
| .           | הויר דOTS, מילון גיבובים                       |
| $x^2$       | הויר אטומט 'x' ^ 2 מילון גיבובים               |
| $R?$        | הויר ביטר R מילון גיבובים                      |
| $R^*$       | הויר סטטוס פונקט פונקט R מילון גיבובים         |
| $R^+$       | הויר סטטוס פונקט פונקט R+ מילון גיבובים        |
| $R_1 R_2$   | הויר סטטוס פונקט פונקט R_1 R_2 מילון גיבובים   |
| $R_1   R_2$ | הויר סטטוס פונקט פונקט R_1   R_2 מילון גיבובים |
| $(R)$       | הויר סטטוס פונקט פונקט (R) מילון גיבובים       |

- \* רצף כמה טרנסיגריה מילון גיבובים
- \* רצף כמה טרנסיגריה מילון גיבובים

## טבלה:

- \* טבלה מילון גיבובים מילון גיבובים ( $\dots, *, ?, +$ )
- \* טבלה מילון גיבובים מילון גיבובים ( $\dots, *, ?, +$ )

## טבולה:

- \* טבולה מילון גיבובים מילון גיבובים ( $\text{letter} = \text{alpha} \dots \text{z}, \text{digit} = \text{0} \dots \text{9}$ )
- \* טבולה מילון גיבובים מילון גיבובים ( $\text{letter} = \text{alpha} \dots \text{z}, \text{digit} = \text{0} \dots \text{9}$ )

## ט�ס אוטומט לאנימציה:

\* טטס אוטומט שפירושו הוא מודול.

\* גודל אוטומט מוגדר כמספר סימני גיבוב, שפה או קבוצת שפות.

## טטס מילויים:

$f : \Sigma \rightarrow \{0, 1\}$

טטס אוטומט מוגדר על ידי סימני גיבוב  $\Sigma$ , סימני מילויים  $\{0, 1\}$  ופונקציית מילוי  $f : \Sigma \rightarrow \{0, 1\}$ .  
רעיון זה מגדיר את מילויי אוטומט  $M$  כפונקציית מילוי  $f$  מ- $\Sigma$  ל- $\{0, 1\}$ .

## טטס מילויים - Tokenizer

\* טטס מילויים מוגדר ב- $\Sigma$  ו- $\{0, 1\}$ .

\* טטס מילויים מוגדר ב- $\Sigma$  ו- $\{0, 1\}$  ופונקציית מילוי  $f : \Sigma \rightarrow \{0, 1\}$ .  
טטס מילויים מוגדר ב- $\Sigma$  ו- $\{0, 1\}$  ופונקציית מילוי  $f : \Sigma \rightarrow \{0, 1\}$ .

!טטס מילויים מוגדר ב- $\Sigma$  ו- $\{0, 1\}$ .

## טטס מילויים נטול סימן סוף:

(DFA)  $M = (\Sigma, Q, \delta, q_0, F)$

$\Sigma$  - סימני גיבוב.

$Q$  - סט של מצבים.

$F \subseteq Q$  - סט של מצבים סופיים.

$\delta : Q \times \Sigma \rightarrow Q$  - פונקציית מילוי.

(NFA)  $M = (\Sigma, Q, \delta, q_0, F)$

$\Sigma$  - סימני גיבוב.

$Q$  - סט של מצבים.

$F \subseteq Q$  - סט של מצבים סופיים.

$\delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q$  - פונקציית מילוי.

\* מילוי אוטומט נס饱ה

ו. מילוי אוטומט  $M$ , מילוי אוטומט נס饱ה  $R$ . קיימת גזירה בין  $M, \omega$  ו-  $R$ .

: NFA- $\Sigma$  נס饱ה אוטומט

$R_1, \dots, R_n$  אוטומטים נס饱ה שמייצגים סדרת מילויים של אוטומט  $M$ .

$R_i$  אוטומט נס饱ה  $M_i$ , NFA- $\Sigma$  : 2 מילוי.

מילים נס饱ה  $M_i$  גזירה : 3 מילוי.

מילים נס饱ה גזירה גזירה : ambiguity גזירה.

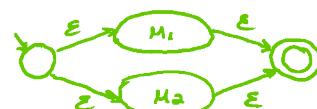
: מילוי אוטומט



$$R = R_1 R_2 *$$



$$R = E *$$



$$R = R_1 | R_2 *$$



$$R = a *$$



$$R = R_1^*$$



$$R = \epsilon *$$

$R_{i+1}$  מילוי גזירה  $\epsilon \rightarrow$  רוחני נס饱ה גזירה ( $R_{i+1} = R_i R_i^*$ ) \*

? מילוי אוטומט?

. מילוי אוטומט CNIC ו- ID ערך.

. מילוי CNIC ו-ID על ידי מילוי אוטומט.

: מילוי אוטומט נס饱ה.

$M_1(\omega)$  נס饱ה -

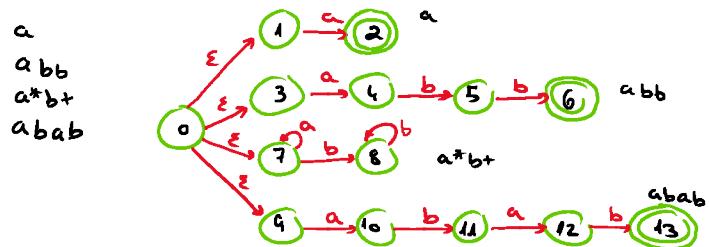
$M_2(\omega)$  נס饱ה -

:

$M_n(\omega)$  נס饱ה -

. מילוי אוטומט נס饱ה גזירה גזירה.

: מילוי אוטומט נס饱ה גזירה גזירה.



## פערין לאינאים:

\* רדיבר:

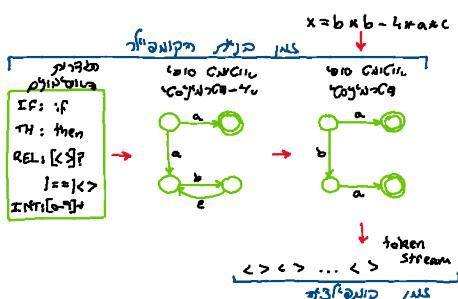
- הוליך הרכבת כויה.
- שרכי צויר - סבב תרשים חדיר גלויים נחים ו/or.

\* אוטומט:

- DFA-L NFA

\* ו/or פ' פ' ערך קלאס אקדמי טקסט מילויים - מילויים בודדים.

## סמן הפעלה:



\* רדיבר סטנדרטם דמויים דינמיים.

\* DFA CNICIN, DFA CNICIN.

\* אוטומט גוף קי', נושא-CNICIN ב-CNICIN לאמת לזרען גוף קי' נושא-CNICIN ב-CNICIN.

הו מושג?

## ט'ז'ט:

\* DFA CNICIN ט'ז'ט צויר קבוצה ט'ז'ט ו/or ט'ז'ט נושא.

שאנו יתרכז על מה שקיים בסיס.

\* ביטריאט DFA ט'ז'ט קבוצת ט'ז'ט ו/or ט'ז'ט הינה נושא.

\* נטוו (קאו שטן נאקו) המבוקש נושא.

\* ערך ט'ז'ט Grune-Lint מילויים נדרשים נושא ב-CNICIN.

## ט'ז'ט ט'ז'ט:

\* הטראנספורמציה של קבוצת ט'ז'ט לאפקט, גם מושג דוגמא ט'ז'ט אפקט ט'ז'ט.

בוחן ט'ז'ט כט'ז'ט, דוחה לאפקט מ-CNICIN.

\* ט'ז'ט: ט'ז'ט ט'ז'ט ט'ז'ט.

## ט'ז'ט CNICIN:

\* הטראנספורמציה של קבוצת CNICIN ט'ז'ט ט'ז'ט.

\* !Flex

\* א"כ אורך גוף קי', ט'ז'ט ט'ז'ט לאפקט ט'ז'ט.

\* טרין: קבוצת דט'ז'ט, ט'ז'ט ט'ז'ט ט'ז'ט, ט'ז'ט ט'ז'ט ט'ז'ט.

\* ט'ז'ט ט'ז'ט.

## עורך מילים גזורה:

- \* גוף בן 3 טרנספורם גזורה.
- \* הוסף אוניברסיטט נטגרט כפונקציית פולינום.
- \* קרייג R: DFA  $\leftarrow$  NFA  $\leftarrow$  אגדת מילים.
- \* פלקס פלט גזורת מילים ב-C/C++.

פלקס פלט גזורת מילים ב-C/C++.

הווריאנטים:

#pragma OpenMP -

SYCL -

## ນິຫາວ ສິນຕັກທີ

9:20 AM Sunday, March 29, 2020

ຄົກລາຄ:

\* ເປົ້າ ຈຸດ ແກ້ໄຂ ທະເລືອກ.

\* ລົບໄວ ແກ້ໄຂ ແກ້ໄຂ ທະເລືອກ

(ກົມ) (Parsing)

: ນິກາ

- ລົບໄວ ແກ້ໄຂ ແກ້ໄຂ ທະເລືອກ ມີຄຸນ ໂດຍ

- ມີ ດາວໂຫຼວງ ທີ່ ຜົນຕະຫຼາກ - ປົນຕະຫຼາກ ຖະແຈ ດີ ດຳ

- ຂົນ ພະຍາຍາ ອີເມວິກ

\* ສົກລົງ

? ມີ ອຽນຄົມ ມີ ທັນ ເພີ້ມ?

? ມີ ຕຣູກ ຢັດ ແກ້ໄຂ

? ມີ ອົງກອນ ທີ່ ອົງກອນ

: ອົງກອນ ດັບ ມີ

$G = \langle V, T, P, S \rangle$

$S \rightarrow aK_a$  . (ພົມພວກ ດັບ) ສົມມົມ -  $V$   
 $K \rightarrow b$  .  
 $K \rightarrow bK$  . ພົມພວກ ດັບ -  $S$

. (token - ພົມພວກ) ສົມມົມ -  $T$

$V \rightarrow (TUV)^*$  : ດັບຕະຫຼາກ. ດັບຕະຫຼາກ. ດັບຕະຫຼາກ -  $P$

? ອົງກອນ ຖະແຈ ຕົວ ຕໍ່ ພົມພວກ?

ເກີຍ, ພົມພວກ ບໍ່ ດັບຕະຫຼາກ ໃຫຍ້ ແລ້ວ ດັບຕະຫຼາກ ຕົວ ຕໍ່ ພົມພວກ.

: ນິກາ

$V = \{S, E\}$   $T = \{id, num, +, *, ;, :=\}$

\*  $S \rightarrow S ; S$

\*  $S \rightarrow id := E$

\*  $E \rightarrow id | num | E + E | E * E | (E)$

$$\begin{array}{l} S \rightarrow S; S \mid id := E \\ E \rightarrow id \mid E + E \mid E * E \mid (E) \end{array}$$

$$\begin{array}{l} x := z; \\ y := x + z; \end{array}$$

$$\begin{array}{l} S \\ S; S \\ id := E; S \\ id := E; id := E \\ id := id; id := E \\ id := id; id := E + id \\ id := id; id := id + id \end{array}$$

$$\begin{array}{l} S \rightarrow S; S \\ S \rightarrow id := E \\ S \rightarrow id := E \\ E \rightarrow \text{empty} \\ E \rightarrow E + E \\ E \rightarrow id \\ E \rightarrow id \end{array}$$

הנחתה כ

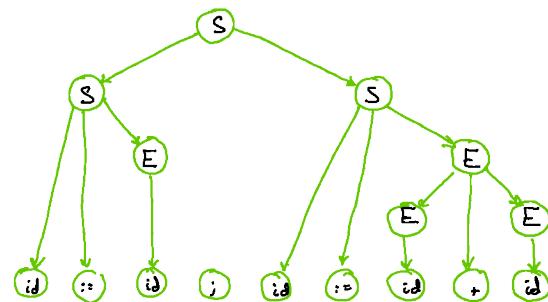
הנחתה היא גורם דרמטי ב證明 תבונה (Derivation) \*

הנחתה היא שפה (Language) \*

הנחתה היא צורה (Sentential Form) \*

הנחתה היא סטרטגיית הוכחה (Proof Strategy)

הנחתה היא רקורסיבית (Recursive)



הנחתה

הנחתה היא מנגנון איסוף נתונים?

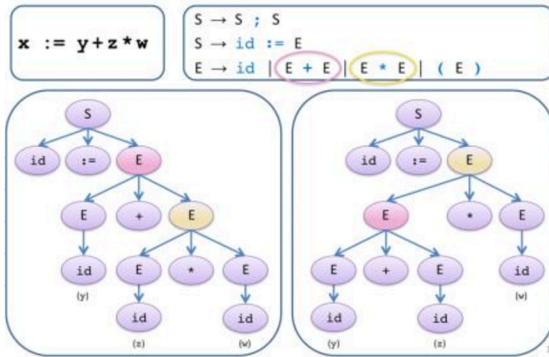
הנחתה היא סדרה של נקודות?

הנחתה היא סדרה של נקודות?

הנחתה היא סדרה של נקודות?

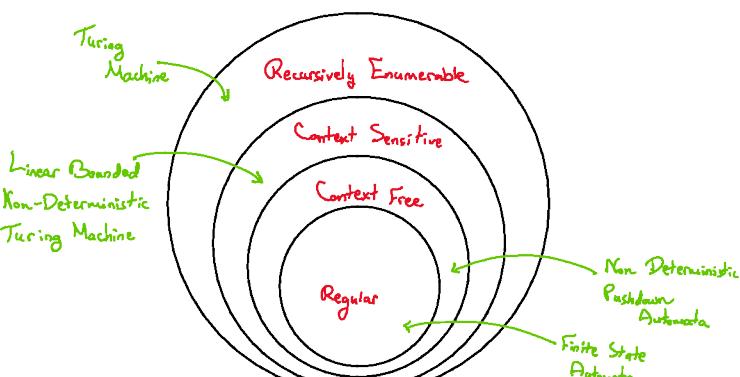
הנחתה היא מנגנון איסוף נתונים: Leftmost Derivation

הנחתה היא סדרה של נקודות: Rightmost Derivation



\* נס סטראטגי מילוי מערך נס סטראטגי נס סטראטגי

## וָיְהִי בְּרֵאשִׁית כֹּל־בְּרֵאשִׁית וְהַדָּבָר כְּלֵל



## וְיֻכָּל בְּרֹאשֵׁת :

\* נס גנרי סטראטגי כטג'יר סיבת: מושך וערוך קומפלקס מעורב לא-טראנסיטיבי

\* Backtracking, First

\* פקטורי מס בעיגול כטג'יר פון:

Non-Deterministic Pushdown Automata!

WTF?!

\* נס גנרי א-טראנסיטיבי כטג'יר גנטיס.

\* נס גנרי טראנסיטיבי כטג'יר גנטיס.

## טַבְּרַכְתָּא לְפָרָא:

\* EARLY-EARLY-PARSING (PARSING) - סטראטגיית פרא-EARLY, סטראטגיית פרא-EARLY-EARLY.

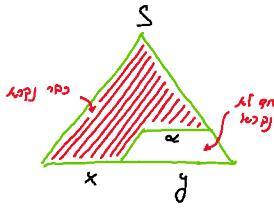
\* גונדריה נס גנרי סטראטגי. נס גנרי סטראטגי (טראנסיטיבי גונדריה נס גנרי)  $O(n^3)$ .

\* נס גנרי סטראטגי סטראטגי. גונדריה מילוי.

רקיינט גונדרי מילוי

מתקני סיביות:

: Top-Down (Predictive) \*

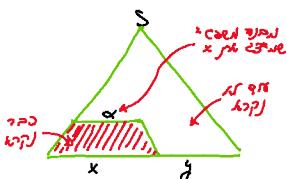


- מתקן סיבית יפה הוא מתקן שמייצג מתקן סיבית יפה.

- "יונס" מתקן סיבית יפה -

- מתקן סיבית יפה מתקן סיבית יפה (בהתאם לדוגמה).

: Bottom-Up (Shift Reduced) \*



- מתקן סיבית יפה הוא מתקן שמייצג מתקן סיבית יפה.

- מתקן סיבית יפה יונס."

- מתקן סיבית יפה מתקן סיבית יפה (בהתאם לדוגמה).

# nihot Top Down

9:31 AM Sunday, April 5, 2020

## Top Down Parsing

ו. ת'ו  $G = \langle V, T, P, S \rangle$  כו' שטח' נ' נ' נ' נ'

•  $G$  לא ו. י' נ' נ' נ'

• נ' נ'

ר' פ' ו' ו' ו' ו' ו'

ו. ו' ו' ו' ו' ו' ו'

ו. ו' ו' ו' ו' ו' ו'

ו. ו' ו' ו' ו' ו'

: (LL) ו' ו' ו' ו'

ו. ו' ו' ו' ו' ו'

ו. ו' ו' ו' ו' ו'

## Recursive Descent Parsing

ו. ו' ו' ו' ו'

ו. ו' ו' ו' ו'

ו. ו' ו' ו' ו'

ו. ו' ו' ו' ו' ו'

```
void match(token t) {
    if (current == t)
        current = next_token();
    else
        error;
}
```

E → LIT | ( E OP E ) | not E  
LIT → true | false  
OP → and | or | xor

פ' ו' ו' ו' ו' ו'

```
void E() {
    if (current ∈ {TRUE, FALSE}) // E → LIT
        LIT();
    else if (current == LPAREN) // E → ( E OP E )
        match(LPAREN); E(); OP(); E(); match(RPAREN);
    else if (current == NOT) // E → not E
        match(NOT); E();
    else
        error;
}

void LIT() {
    if (current == TRUE) match(TRUE);
    else if (current == FALSE) match(FALSE);
    else error;
}

void OP() {
    if (current == AND) match(AND);
    else if (current == OR) match(OR);
    else if (current == XOR) match(XOR);
    else error;
}
```

## ריצוף אובייקטיבי:

```

Node E() {
    result = new Node();
    result.name = "E";
    if (current ∈ {TRUE, FALSE}) // E → LIT
        result.addChild(LIT());
    else if (current == LPAREN) // E → ( E OP E )
        result.addChild(match(LPAREN));
    result.addChild(E());
    result.addChild(OP());
    result.addChild(E());
    result.addChild(match(RPAREN));
    else if (current == NOT) // E → not E
        result.addChild(match(NOT));
    result.addChild(E());
    else error;
    return result;
}

```

ריצוף אובייקטיבי מילויים של סימני הפעלה

הנתקה מ- $E \rightarrow ( E \text{ OP } E )$  ו- $E \rightarrow \text{NOT } E$

Node מילויים של סימני הפעלה

obj מילויים של Node

.obj מילויים של obj

## : Recursive Descent

```

void A() {
    choose an A-production, A → X1X2...Xk;
    for (i = 1; i ≤ k; i++) {
        if (Xi is a nonterminal) call procedure Xi();
        else if (Xi == current) current = next_token();
        else error;
    }
}

```

ריצוף אובייקטיבי מילויים של סימני הפעלה

backtracking (3c) | backtracking (3d) | backtracking (3e)

.backtracking -> end: if backtracking (3f)

## : סידור

$E \rightarrow \text{LIT} \mid ( E \text{ OP } E ) \mid \text{NOT } E$

מילויים של סימני הפעלה

מילויים של סימני הפעלה  $E \rightarrow ( E )$

מילויים של סימני הפעלה  $E \rightarrow E \text{ OP } E$

## : LL(k)

LL(k) מילויים של סימני הפעלה מילויים של סימני הפעלה

.Top-Down ריצוף

(k) מילויים של סימני הפעלה

(k) מילויים של סימני הפעלה

.מילויים של סימני הפעלה backtracking, מילויים של סימני הפעלה

.LL(k) מילויים של סימני הפעלה מילויים של סימני הפעלה

.LL(k) מילויים של סימני הפעלה מילויים של סימני הפעלה

## : First Sets

ל- $A \rightarrow \alpha$  מילויים של סימני הפעלה מילויים של סימני הפעלה

.FIRST( $\alpha$ ) מילויים של סימני הפעלה מילויים של סימני הפעלה

,  $A \rightarrow \alpha$  מילויים של סימני הפעלה

. $\alpha = e$  מילויים של סימני הפעלה  $G = \text{First}(e)$

. $\alpha = f$  מילויים של סימני הפעלה מילויים של סימני הפעלה

•  $\text{First}(E) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } \alpha = \text{label of some transition from } q_0 \text{ to } q_f\}$

•  $\text{First}(A) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } A \rightarrow^* S \text{ and } \alpha \in \text{label of some transition from } q_0 \text{ to } q_f\}$

( $A$  is nullable)  $A \rightarrow^* E \Rightarrow \text{First}(A) \subseteq \text{First}(E)$

: תרשים

$\text{First}(LIT) = \{\text{true}, \text{false}\}$

$\text{First}(EOP) = \{', \}$

$\text{First}(\text{not } E) = \{\text{not}\}$

$E \rightarrow LIT \mid (EOP) \mid \text{not } E$

$LIT \rightarrow \text{true} \mid \text{false}$

$OP \rightarrow \text{and} \mid \text{or} \mid \text{xor}$

$E \rightarrow \alpha \Rightarrow \text{First} \rightarrow \text{DP} \rightarrow \alpha$

• תרשים של אוניברסלי First

•  $\text{First}(E) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } E \rightarrow^* S \text{ and } \alpha \in \text{label of some transition from } q_0 \text{ to } q_f\}$

•  $\text{First}(E) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } E \rightarrow^* S \text{ and } \alpha \in \text{label of some transition from } q_0 \text{ to } q_f\}$

$LL(1) \subset LL(2) \subset LL(3) \subset \dots$

: תרשים

$\text{First} \subseteq \sum : LL(1)$

$\text{First} \subseteq \sum \times \sum : LL(2)$

•  $\text{First}(E) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } E \rightarrow^* S \text{ and } \alpha \in \text{label of some transition from } q_0 \text{ to } q_f\}$

(nullable)  $\text{First}(A) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } A \rightarrow^* S \text{ and } \alpha \in \text{label of some transition from } q_0 \text{ to } q_f\}$

: תרשים Recursive Descent

```
void A() {  
    find an A-production,  $A \rightarrow \overbrace{X_1 X_2 \dots X_k}^a$ ,  
    such that current  $\in FIRST(a)$   
    for (i = 1; i <= k; i++) {  
        if ( $X_i$  is a nonterminal) call procedure  $X_i()$ ;  
        else if ( $X_i == \text{current}$ ) current = next_token();  
        else error;  
    }  
}
```

: Follow Sets

? פירסום (nullable)  $\text{Follow}(S) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } S \rightarrow^* A \text{ and } A \rightarrow^* \alpha\}$

$S \rightarrow AB \mid C \quad A \rightarrow a \mid \epsilon \quad B \rightarrow b \quad : \text{len}^*$

•  $\text{Follow}(A) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } S \rightarrow^* A \text{ and } A \rightarrow^* \alpha\}$

• תרשים Follow

: A תרשים LR

•  $\text{Follow}(A) = \{\alpha \mid \text{there exists } S \in \text{NFA such that } S \rightarrow^* A \text{ and } A \rightarrow^* \alpha\}$

## :pged zew Recursive Descent

```

void A() {
     $\overbrace{a}$ 
    find an A-production,  $A \rightarrow \overbrace{X_1 X_2 \dots X_k}$ ,
    such that current  $\in \text{FIRST}(a)$ 
    or  $a \in \text{FIRST}(a)$  and current  $\in \text{FOLLOW}(A)$ 
    for (i = 1; i < k; i++) {
        if ( $X_i$  is a nonterminal) call procedure  $X_i()$ ;
        else if ( $X_i == \text{current}$ ) current = next_token();
        else error;
    }
}

```

## :LL(0)-ה א'ג'ז'יר

$\text{term} \rightarrow \text{ID} \mid \text{indexed}$  :ר'ג'ז'ב'ה נ'ג'ז'

$\text{indexed} \rightarrow \text{ID} [\text{expr}]$

$$\left. \begin{array}{l} \text{term} \rightarrow \text{ID} \\ \text{term} \rightarrow \text{indexed} \end{array} \right\} \text{First (ID)} = \{\text{ID}\}$$

$$\left. \begin{array}{l} \text{term} \rightarrow \text{ID} \\ \text{term} \rightarrow \text{indexed} \end{array} \right\} \text{First (indexed)} = \{\text{ID}\}$$

! LL(0) י'ג'ז'ס ר'ג'ז'ב'ה First/First א'ג'ז'יר ו'ז'

## :ה'ג'ז'ר כ'ג'ז'ב'ה

. ל'ג'ז'ר כ'ג'ז'ב'ה ר'ג'ז'ב'ה First/First א'ג'ז'יר ו'ז'

? LL(0) י'ג'ז'ס ר'ג'ז'ב'ה First/First א'ג'ז'יר ו'ז'

## :(ג'ז'ר כ'ג'ז'ב'ה) Left Factoring

:ל'ג'ז'ר כ'ג'ז'ב'ה First/First א'ג'ז'יר ו'ז'

$\text{term} \rightarrow \text{ID} \mid \text{indexed}$   
 $\text{indexed} \rightarrow \text{ID} [\text{expr}]$

$\downarrow$

$\text{term} \rightarrow \text{ID} \cdot \text{after-ID}$   
 $\text{after-ID} \rightarrow [\text{expr}] \Sigma$

$x(y+2) \rightarrow x y + x 2$  ו'ז'ג'ז'ב'ה ר'ג'ז'ב'ה First/First א'ג'ז'יר ו'ז'

:ר'ג'ז'ב'ה First/First א'ג'ז'יר ו'ז'

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$   
 $\quad | \text{ if } E \text{ then } S$   
 $\quad | T$

$S \rightarrow \text{if } E \text{ then } S \text{ S}'$   
 $\quad | T$   
 $S' \rightarrow \text{else } S \mid \epsilon$