

تحليل النص: الإعراب

**SYNTHETIC ANALYSIS:
PARSING**

المحتويات TOPICS

Context free grammars نحو حر/مستقل-عن السياق/التعبير

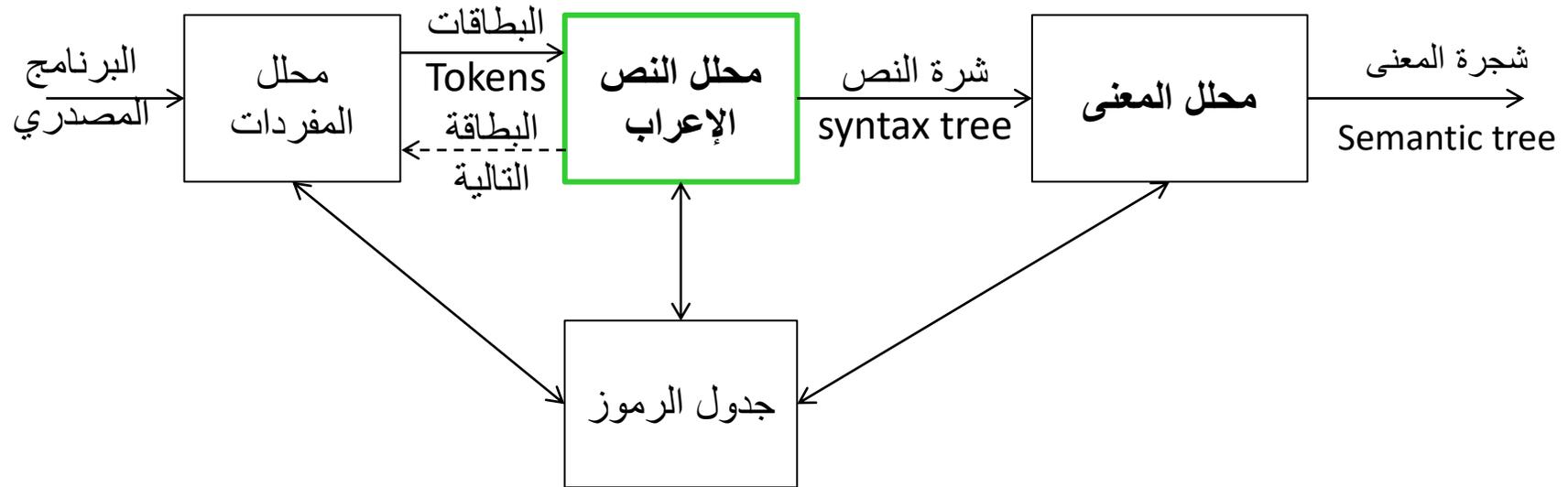
Derivations اشتقاق

Parse Trees شجرة الإعراب

Ambiguity الالتباس

Precedence أسبقية

تحليل المعنى - Parsing



- طور تحليل المعنى يعتمد على مفردات النص ليكشف معناها من خلال توافقها مع قواعد لغة البرمجة
- قواعد اللغة يعبر عنها بتشكيل ما يعرف بتشكيل "النحو مستقل/حر السياق"

Context-free grammar (BNF (Backus-Naur Form))

- هو يستخدم لوصف قواعد صياغة نص اللغة (النحو grammar)
- والنحو يصف أسبقيات وأولويات بناء اللغات hierarchical structure
- مثال:
- `if (expression) statement else statement`
- جملة `if` عبارة عن لصق concatenation للكلمة المحجوزة `if`, قوس فتح (, تعبير, قوس قفل (, جملة, الكلمة المحجوزة `else`, و جملة أخرى.

• **المثال:**

if (expression) statement **else** statement

- باستخدام المتغير *expr* للتعبير expression, والمتغير *stmt* للجملة statement
يمكن وصف قاعدة if كالتالي:

$stmt \rightarrow \mathbf{if} (expr) stmt \mathbf{else} stmt$

- السهم \rightarrow يقرأ "لها الشكل form": *stmt* لها الشكل if تعبير جملة else جملة
وتسمى هذه القاعدة النحوية بمنتج a production
- المفردات if, (,), else تسمى بالطرفيات terminals: مفردة تأولها نهائي
- المفردات *stmt*, *expr* تسمى باللاطرفيات nonterminals: غير منتهية التأويل

مكونات القواعد النحوية حرة السياق

• تتكون من:

1. فئة من الرموز الطرفية terminal symbols
• وهي تلك البطاقات tokens, والتي تمثل المفردات الأساسية للغة
2. فئة من لا طرفيات nonterminals, وتسمى syntactic variable المتغير النحوي
• وهو يعبر عن صياغة تشمل فئة من مفردات طرفية
3. المنتجات productions وهي تصف أسلوب/أساليب تركيب القاعدة النحوية:
• بها لا طرفية تسمى الرأس head أو الجانب الأيسر left side, سهم,
• وسلسلة طرفيات أو لا طرفيات تسمى الجسم body أو الطرف الأيمن
4. رمز البداية start symbol: وهو أحد لا طرفيات وغالباً تبدأ به المنتجات

مكونات القواعد النحوية حرة السياق

- المجموعات

- منتجات productions تحمل نفس اسم لاطرفية الرئيسية head , يتم تجميع

أجسامها في مجموعة يفصل بين عناصرها بالرمز | أو

- مثال: لدينا مجموعة بمنتجات قاعدة نحوية:

$list \rightarrow list + digit$

$list \rightarrow list - digit$

$list \rightarrow digit$

$digit \rightarrow 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

- المجموعة $list$ يمكن أن تكتب:

$list \rightarrow list + digit | list - digit | digit$

مكونات القواعد النحوية حرة السياق

- مثال آخر: منتجات productions نحوية التعبير الرياضي

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\langle \text{expr} \rangle \rightarrow \text{id}$
 $\langle \text{expr} \rangle \rightarrow \text{num}$
 $\langle \text{op} \rangle \rightarrow +$
 $\langle \text{op} \rangle \rightarrow -$
 $\langle \text{op} \rangle \rightarrow *$
 $\langle \text{op} \rangle \rightarrow /$

- ماهي الرموز الطرفية **terminals**؟
the terminal symbols are: *num, id + - * /*

- ماهي الرموز لا الطرفية **nonterminals**؟

The nonterminal symbols are $\langle \text{expr} \rangle$ and $\langle \text{op} \rangle$,
and $\langle \text{expr} \rangle$ is the start symbol كما تبدأ القاعدة بلا طرفي

مكونات القواعد النحوية حرة السياق

- منتجات productions نحوية التعبير الرياضي

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\langle \text{expr} \rangle \rightarrow \text{id}$
 $\langle \text{expr} \rangle \rightarrow \text{num}$
 $\langle \text{op} \rangle \rightarrow +$
 $\langle \text{op} \rangle \rightarrow -$
 $\langle \text{op} \rangle \rightarrow *$
 $\langle \text{op} \rangle \rightarrow /$

- ماهي المجموعات؟

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$
 $\langle \text{op} \rangle \rightarrow + \mid - \mid * \mid /$

المجموعات هي اختصار لوصف القواعد

- هو استخلاص تعريف تام لنص مبني على قاعدة نحوية,
- ويكون ذلك باستبدال لا طرفي nonterminal بجسم body أحد منتجات production ذلك لا الطرفي
- مع البداية برمز البداية start symbol
- وتكرار عملية الاستبدال حتى يتم تعريف جملة النص كلها
- ✓ أي يصبح جميع عناصرها طرفيات terminals

DERIVATION

الاشتقاق

- مثال: باستخدام المنتجات productions

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\langle \text{expr} \rangle \rightarrow \text{id}$

$\langle \text{op} \rangle \rightarrow *$

- ما هو اشتقاق التعبير $X * Y$ الممثل بسلسلة البطاقات $\langle \text{id}, 1 \rangle \langle * \rangle \langle \text{id}, 2 \rangle$

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

- تقرأ: expr تشتق derives التعبير expr op expr

$\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} * \langle \text{expr} \rangle$

$\Rightarrow \text{id} * \text{id}$

- وتسمى اشتقاق $\text{id} * \text{id}$ من expr

• مثال: باستخدام المنتجات productions $\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\langle \text{expr} \rangle \rightarrow \text{id}$

$\langle \text{op} \rangle \rightarrow *$

• الاشتقاق للتعبير $X - Y$ الممثل بالبطاقات $\langle \text{id}, 1 \rangle \leftrightarrow \langle \text{id}, 2 \rangle$

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} * \langle \text{expr} \rangle$

Error

• لا يمكن اشتقاق $\text{id} - \text{id}$ من نحو القاعدة expr

الاشتقاق

• مثال: باستخدام المنتجات productions

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$

$\langle \text{op} \rangle \rightarrow + \mid - \mid * \mid /$

• الاشتقاق للتعبير $X - Y$ الممثل بالبطاقات $\langle \text{id}, 1 \rangle \leftrightarrow \langle \text{id}, 2 \rangle$

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} - \langle \text{expr} \rangle$

$\Rightarrow \text{id} - \text{id}$

• تم اشتقاق $\text{id} - \text{id}$ من expr

الاشتقاق

• مثال: باستخدام المنتجات productions

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$

$\langle \text{op} \rangle \rightarrow + \mid - \mid * \mid / \mid =$

بالبطاقات $\langle \text{id}, 4 \rangle \Leftrightarrow \langle \text{id}, 1 \rangle \langle / \rangle \langle \text{id}, 2 \rangle$ وهي تعبر عن $\text{var} = x / y$

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} = \langle \text{expr} \rangle$

$\Rightarrow \text{id} = \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} = \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} = \text{id} / \langle \text{expr} \rangle$

$\Rightarrow \text{id} = \text{id} / \text{id}$

DERIVATION

من مبادئ الاشتقاق

- رمز الاشتقاق \Rightarrow
- وتقرأ جملة الاشتقاق من اليسار إلى اليمين
- $A \Rightarrow \gamma$ وتقرأ: A تشتق γ "جاما" في خطوة واحدة

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$

نعرب البطاقة $\langle \text{id}, 2 \rangle$ وهي قد تعبر عن متغير في جملة شرط مثل `if (active)`

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle$

$\Rightarrow \text{id}$

• $A \Rightarrow^* \delta$ وتعني: A تشتق δ "دلتا" في صفر أو أكثر خطوات

$\langle \text{term} \rangle \rightarrow \text{id}$

نحو يعرب في لاختوة للبطاقة $\langle \text{id}, 2 \rangle$ وهي قد تعبر عن متغير في جملة شرط مثل `if (active)`

$\langle \text{term} \rangle \Rightarrow \text{id}$

نحو يعرب في خطوات متعددة لنفس البطاقة

$\langle \text{expr} \rangle \rightarrow \langle \text{exp} \rangle \mid \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \langle \text{term} \rangle$

$\langle \text{term} \rangle \rightarrow \text{id} \mid \text{num}$

$\langle \text{op} \rangle \rightarrow + \mid - \mid * \mid / \mid =$

$\langle \text{expr} \rangle \Rightarrow \langle \text{exp} \rangle$

$\langle \text{expr} \rangle \Rightarrow \langle \text{term} \rangle$

$\Rightarrow \text{id}$

• $A \Rightarrow^+ \chi$ وتعني: A تشتق χ "تساي" في واحد أو أكثر خطوات

• إذا كانت لدينا $\alpha \Rightarrow^* \beta$ و $\beta \Rightarrow \delta$ فإن $\alpha \Rightarrow^* \delta$

إذا كانت لدينا $\langle \text{expr} \rangle \Rightarrow^* \langle \text{term} \rangle$

وإذا كانت لدينا $\langle \text{term} \rangle \Rightarrow \text{id}$

فإن $\langle \text{expr} \rangle \Rightarrow^* \text{id}$

- إذا كان $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n$, فإننا نقول " α_1 تشتق α_n " في n خطوات
- إذا كان $S \Rightarrow \alpha$ و α يمكن أن تحتوي على لاطرفيات nonterminals, عندها تسمى α بشبه جملة **sentential**.
- إذا كان α لا تحتوي على لاطرفيات nonterminals, عندها تسمى α جملة **sentence**.

$$E \rightarrow E A E \mid (E) \mid -E \mid id$$

$$A \rightarrow + \mid - \mid * \mid / \mid \uparrow$$

- النص $(id+id) -$ هو جملة **sentence** على قواعد الإعراب السابقة لأن:

$$E \Rightarrow -E \Rightarrow -(E) \Rightarrow -(EAE) \Rightarrow$$

$$-(\mathbf{id}AE) \Rightarrow -(\mathbf{id}+E) \Rightarrow -(\mathbf{id}+\mathbf{id})$$

$$E \Rightarrow -(\mathbf{id}+\mathbf{id}) \quad \text{فنكتب:}$$

وتسمى: اشتقاق $(\mathbf{id}+\mathbf{id}) -$ من E

من مبادئ الاشتقاق

DERIVATION

G: grammar النحو

S: start symbol رمز البداية

L(G): the language generated by **G** لغة نتجت من النحو

- النصوص strings في اللغة **L(G)** ممكن لها أن تحتوي فقط على رموز طرفية terminal symbols مبنية حسب النحو **G**.
- سلسلة الطرفيات المكونة للنص **w** تعتبر من اللغة **L(G)** فقط إذا كان $S \Rightarrow w$.
- النص **w** يسمى جملة بالنحو **G**.
- أي لغة تنتج من خلال قواعد نحوية فهي تسمى "لغة مستقلة السياق context-free language".
- إذا وجد قواعد نحوية مختلفة G_1, G_2 تنتج نفس اللغة, تسمى قواعد نحوية متكافئة equivalent.

موضوعنا التالي:

تحليل النص: يستمر

شجرة الإعراب

PARSE TREE

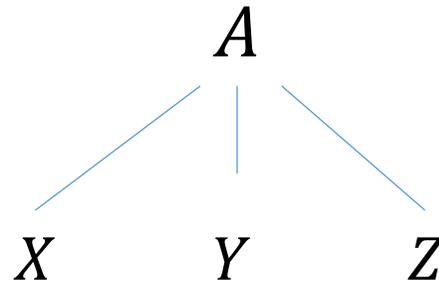
الالتباس

AMBIGUITY

- شجرة الإعراب تعرض خريطة لكيف يقوم رمزُ بداية القاعدة النحوية باشتقاق نصاً في اللغة

$$A \rightarrow XYZ$$

- شجرة الإعراب للاطرفي A nonterminal يكون لها عقدة داخلية interior node بالاسم A مع ثلاث أطفال $X, Y, and Z$ من اليسار إلى اليمين



PARSE TREE

شجرة الإعراب

- عناصر شجرة الإعراب:

1. الجذر root ويحمل اسم رمز البداية is labeled by the start symbol
2. عقد داخلية interior nodes كل منها مسماة labeled بلاطرفية
3. أوراق leaves مسماة labeled بطرفيات terminals وهي عقد بلا أطفال

- شجرة نص إعراب التعبير 9-5+2 المشتق من منتجات القاعدة النحوية:

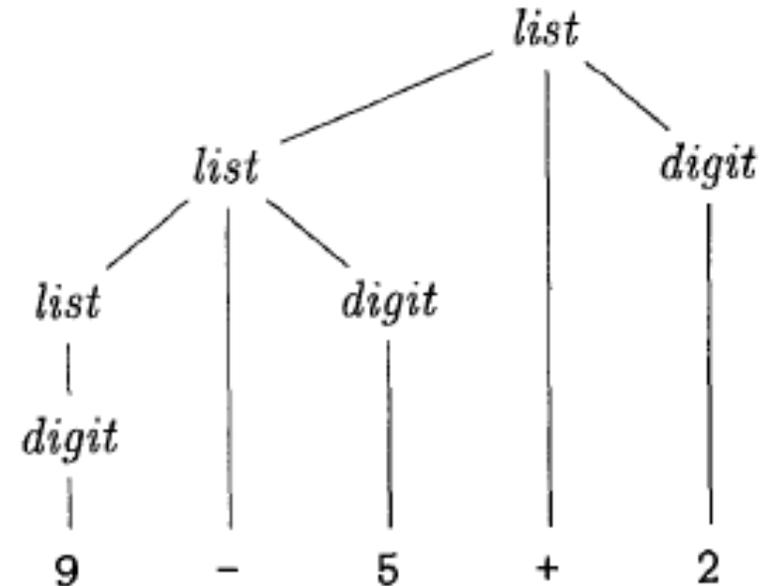
منتجات Productions

$list \rightarrow list + digit$

$list \rightarrow list - digit$

$list \rightarrow digit$

$digit \rightarrow 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$



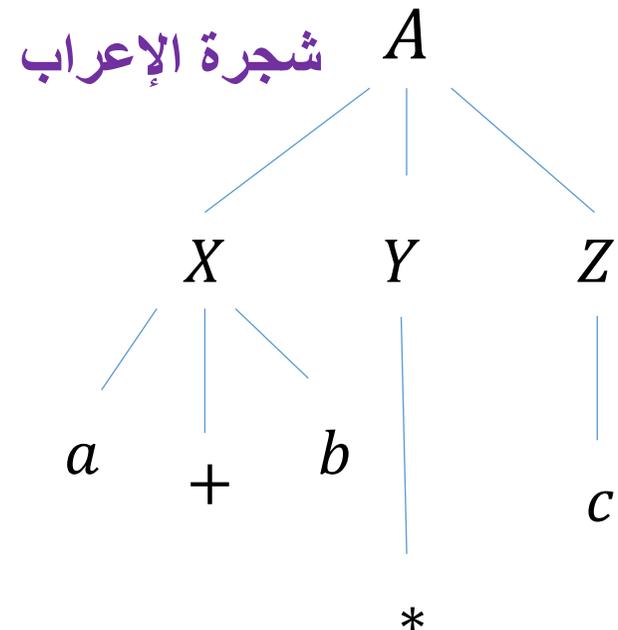
STRING PARSING

إعراب النص

- عملية بناء شجرة الإعراب لنص ما تسمى إعراب النص string parsing
- أولوية أوراق شجرة النص من اليسار إلى اليمين
- إذا كانت X و Y طفلين من نفس الوالد, و X تقع يسار Y
- عندها أطفال وأحفاد X يقعوا على يسار Y

جملة ما
 $A = a + b * c$

قاعدة نحوية
 $A \rightarrow X Y Z$
 $X \rightarrow a + b$
 $Y \rightarrow *$
 $Z \rightarrow c$



هذا الترتيب يسمى بالاشتقاق أقصى اليسار

Leftmost derivation

STRING PARSING

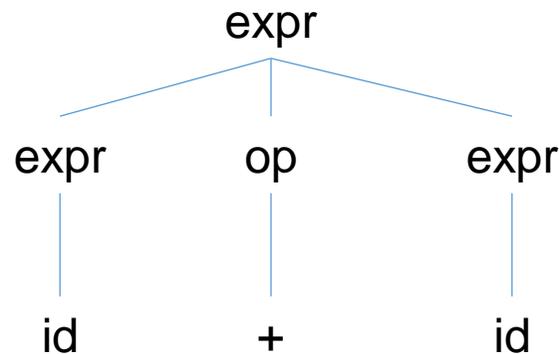
إعراب النص

- س: ماهي شجرة إعراب النحوية التالية للسلسلة: id + id ?

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\langle \text{expr} \rangle \rightarrow \text{id}$

$\langle \text{op} \rangle \rightarrow +$



STRING PARSING

إعراب النص

- س: ماهو الاشتقاق وشجرة إعراب النحوية التالية للتعبير الرياضي المصاحب؟

Grammar:

$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$

$\langle \text{op} \rangle ::= + \mid - \mid * \mid /$

Expression:

$x + 2 * y$

Derivation:

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} * \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} * \text{id}$

STRING PARSING

شجرة إعراب النص

Expression:

$x + 2 * y$

Derivation:

$\langle \text{expr} \rangle \Rightarrow \text{expr} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} + \langle \text{expr} \rangle$

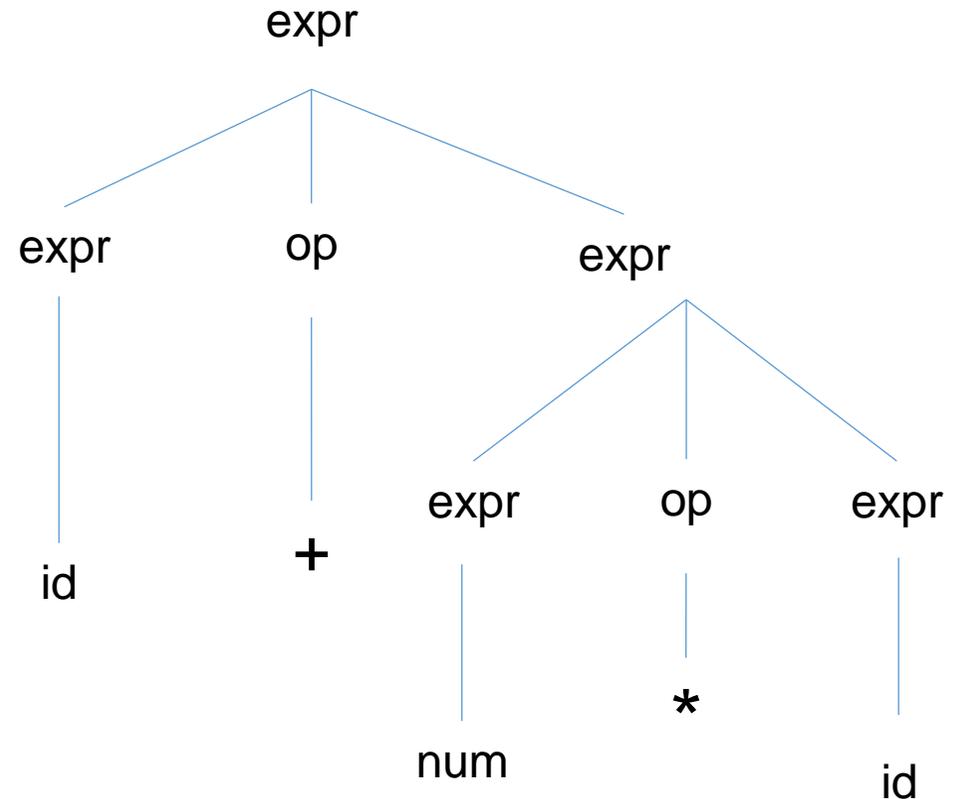
$\Rightarrow \text{id} + \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} + \text{num} \langle \text{op} \rangle \langle \text{expr} \rangle$

$\Rightarrow \text{id} + \text{num} * \langle \text{expr} \rangle$

$\Rightarrow \text{id} + \text{num} * \text{id}$

Parsing tree:



A SECOND DERIVATION!

اشتقاق آخر!

Grammar:

$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$
 $\langle \text{op} \rangle ::= + \mid - \mid * \mid /$

Expression:

$x + 2 * y$

Derivation:

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \underline{\langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle} \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} * \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} * \text{id}$

TWO PARSING TREES!

شجرة إعراب أخرى!

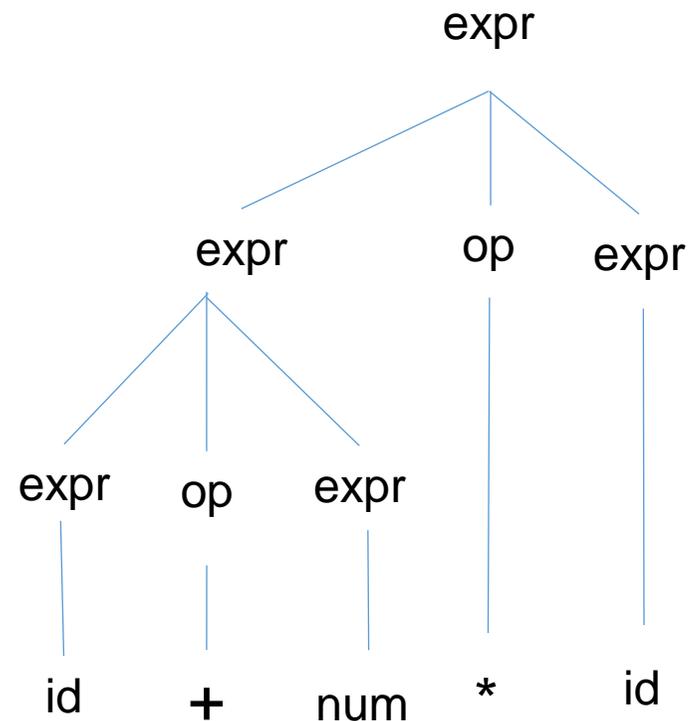
Expression:

$x + 2 * y$

Derivation 2:

$\langle \text{expr} \rangle \Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} \langle \text{op} \rangle \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} \langle \text{op} \rangle \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} * \langle \text{expr} \rangle$
 $\Rightarrow \text{id} + \text{num} * \text{id}$

Parsing tree 2:

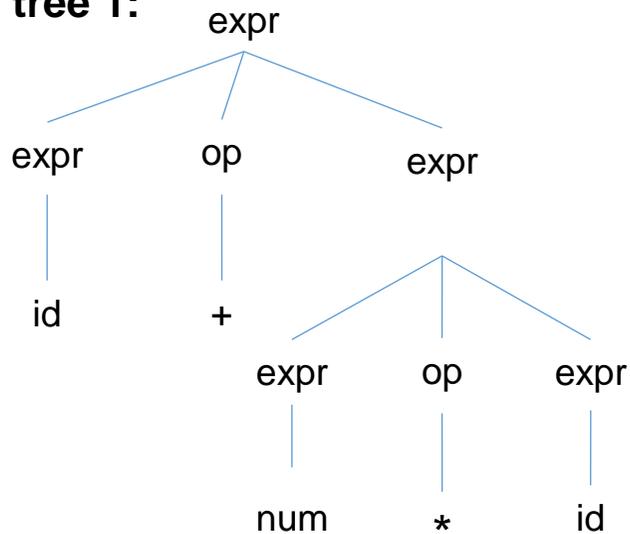


STRING PARSING

إعراب النص

• س: هل من فرق بين شجرتي الإعراب 1 و 2؟

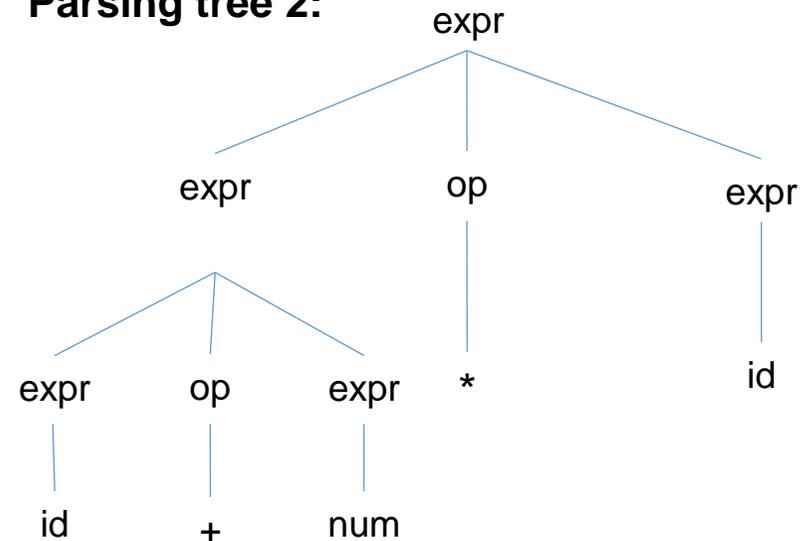
Parsing tree 1:



تكافئ:

✓ id+(num*id)

Parsing tree 2:



تكافئ:

✗ (id+num)*id

Grammar:

$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$
 $\langle \text{op} \rangle ::= + \mid - \mid * \mid /$

Expression:

$x + 2 * y$

- النحو Grammar الذي ينتج أكثر من شجرة إعراب واحدة لجملة ما يسمى نحو ملتبس/غامض/مبهم **ambiguous**.

النحو Grammar:

$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle \mid \text{id} \mid \text{num}$
 $\langle \text{op} \rangle ::= + \mid - \mid * \mid /$

جملة تعبير Expression:

$x + 2 * y$

تكافئ:

✓ $\text{id} + (\text{num} * \text{id})$

تكافئ:

✗ $(\text{id} + \text{num}) * \text{id}$

• باعتبار نحو الصيغة $\langle \text{stmt} \rangle$:statement

```
 $\langle \text{stmt} \rangle ::=$  if  $\langle \text{expr} \rangle$  then  $\langle \text{stmt} \rangle$   
| if  $\langle \text{expr} \rangle$  then  $\langle \text{stmt} \rangle$  else  $\langle \text{stmt} \rangle$   
| other stmts
```

• والجملة:

if E_1 then if E_2 then S_1 else S_2

AMBIGUITY

مثال للالتباس

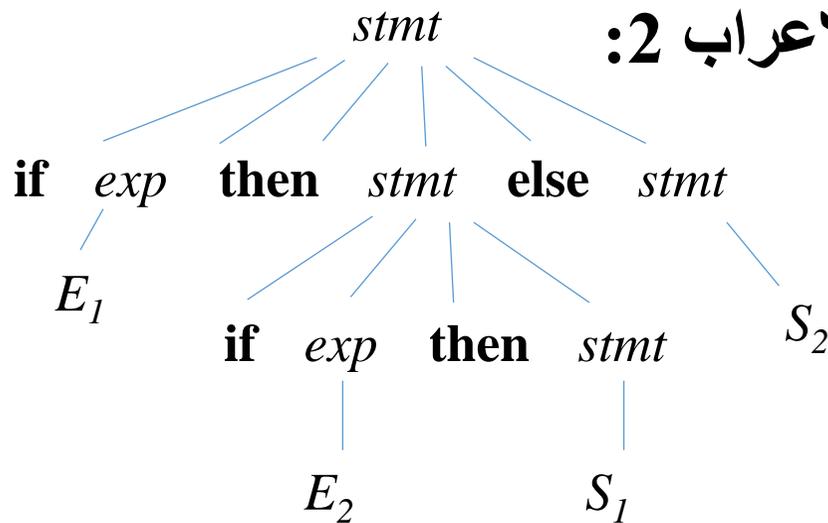
• الجملة

والنحو

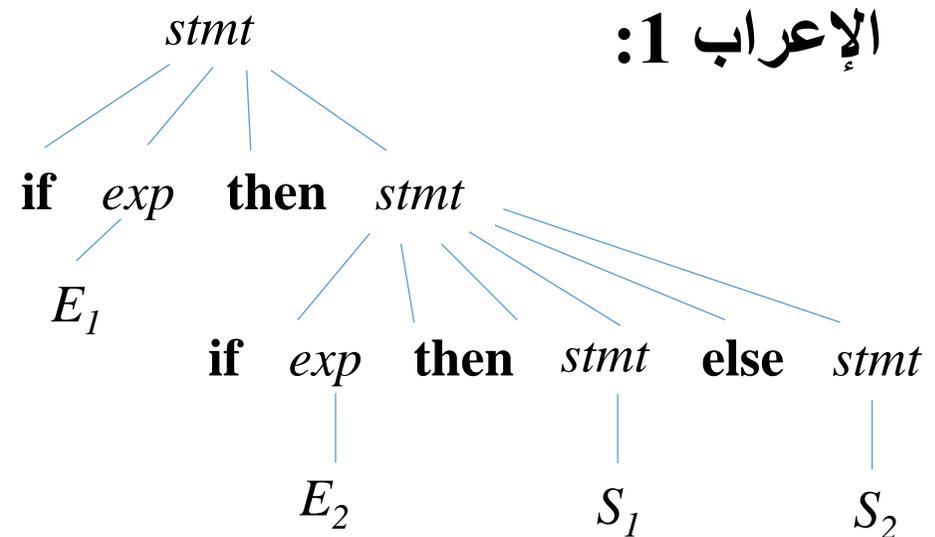
$\langle \text{stmt} \rangle ::=$ if $\langle \text{expr} \rangle$ then $\langle \text{stmt} \rangle$
 |
 | if $\langle \text{expr} \rangle$ then $\langle \text{stmt} \rangle$ else $\langle \text{stmt} \rangle$
 |
 | other stmts

if E_1 then if E_2 then S_1 else S_2

الإعراب 2:



• الإعراب 1:



• ماهو الالتباس بين الإعرابين؟

الإعراب 1: الجملة S_1 تنفذ فقط إذا تحقق تعبير الشرط E_1 وتحقق الشرط E_2 أيضاً، أما الجملة S_2 تنفذ فقط إذا تحقق تعبير الشرط E_1 ولم يتحقق الشرط E_2

الإعراب 2: الجملة S_1 تنفذ فقط إذا تحقق تعبير الشرط E_1 وتحقق الشرط E_2 أيضاً، أما الجملة S_2 تنفذ فقط إذا لم يتحقق تعبير الشرط E_1

- القواعد النحوية الملتبسة كل منها قد يمكن إلغاء الالتباس فيها بأسلوب مختلف
- مثلاً: استخدام تطبيق أسلوب فرض **الاسبقيات precedence** لحذف الالتباس من القواعد النحوية الخاصة بالتعبيرات المحتوية على العمليات الرياضية + , - , * و ./.

الاسبقيات precedence

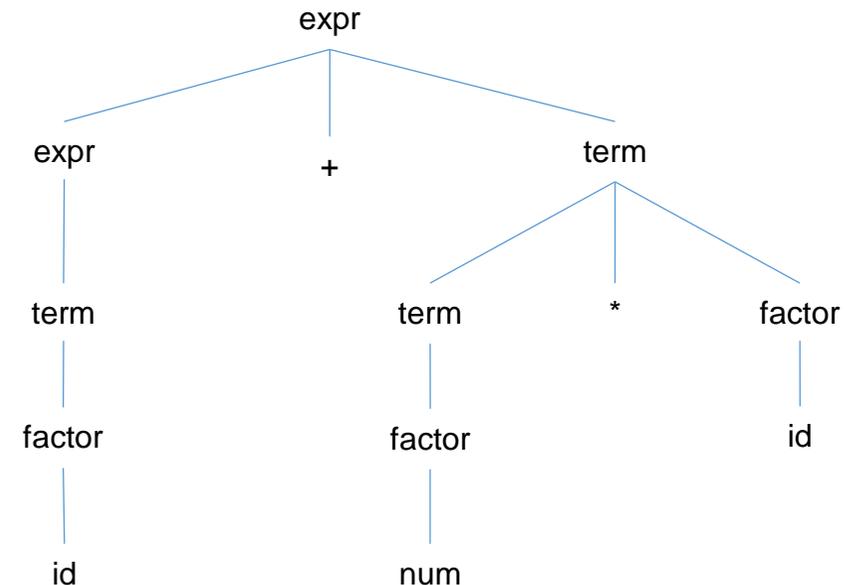
	$\langle \text{expr} \rangle ::= \langle \text{expr} \rangle + \langle \text{term} \rangle$
	$\quad \quad \quad \langle \text{expr} \rangle - \langle \text{term} \rangle$
	$\quad \quad \quad \langle \text{term} \rangle$
نقسم القواعد إلى مجموعات مرتبة حسب أسبقيات استدعائها	$\langle \text{term} \rangle ::= \langle \text{term} \rangle * \langle \text{factor} \rangle$
أسبقيات تضمن تنفيذها السليم	$\quad \quad \quad \langle \text{term} \rangle / \langle \text{factor} \rangle$
لا تصل لمنتج term إلا عبر expr	$\quad \quad \quad \langle \text{factor} \rangle$
لا تصل لمنتج factor إلا عبر term	$\langle \text{factor} \rangle ::= \text{num}$
	$\quad \quad \quad \text{id}$

ELIMINATING AMBIGUITY

حذف الالتباس

- الآن الجملة $id + num * id$ لها فقط اشتقاق واحد أقصى يساري leftmost

$\langle expr \rangle \Rightarrow \langle expr \rangle + \langle term \rangle$
 $\Rightarrow \langle term \rangle + \langle term \rangle$
 $\Rightarrow \langle factor \rangle + \langle term \rangle$
 $\Rightarrow id + \langle term \rangle$
 $\Rightarrow id + \langle term \rangle * \langle factor \rangle$
 $\Rightarrow id + \langle factor \rangle * \langle factor \rangle$
 $\Rightarrow id + num * \langle factor \rangle$
 $\Rightarrow id + num * id$



Grammar:

$\langle expr \rangle ::= \langle expr \rangle + \langle term \rangle \mid \langle expr \rangle - \langle term \rangle \mid \langle term \rangle$
 $\langle term \rangle ::= \langle term \rangle * \langle factor \rangle \mid \langle term \rangle / \langle factor \rangle \mid \langle factor \rangle$
 $\langle factor \rangle ::= num \mid id$

حذف الالتباس من جمل الاختبار **if**

- إعادة صياغة منتجات قواعد جمل **if** النحوية لتجنب الالتباس في اقتران أمر **else** بأمر **then** المناسب, *match else with then*
- اقتران كل أمر **else** مع أقرب أمر **then** غير مقترن **unmatched**
- وهو ما ينويه المبرمج عادةً. **قاعدة نحوية غير ملتبسة لجملة **if****

<stmt> ::= <matched> مقترنة

| <unmatched> غير مقترنة

<matched> ::= if (<expr>) then <matched> else <matched>

| other stmts جمل أخرى داخل جملة الشرط

<unmatched> ::= if (<expr>) then < stmt > بلا عكس الشرط

| if (<expr>) then <matched> else <unmatched>

جملة **else** غير مرتبطة

حذف الالتباس من جمل الاختبار **if**

مثال: جزء من برنامج مصدري به التعابير الشرطية,

```
if (x==0)
    if (y==0)
        z = m;
    else
        z = g;
```

البطاقات tokens التي نحصل عليها بعد تحليل المفردات:

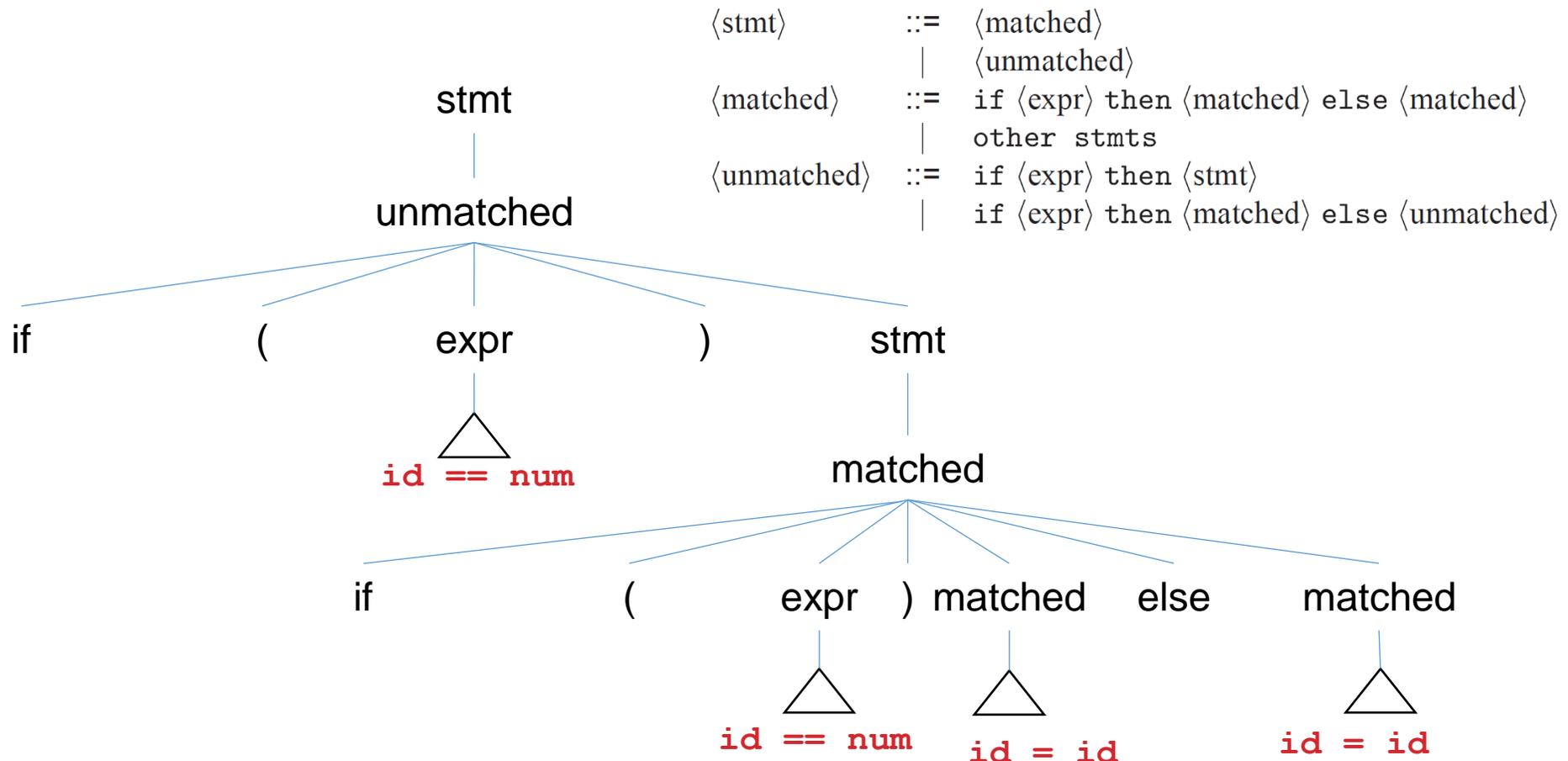
```
if ( id == num ) if (id == num) id = id ; else id = id ;
```

ELIMINATING AMBIGUITY

حذف الالتباس من جمل الاختبار if

البطاقات tokens التي نحصل عليها بعد تحليل المفردات:

if (id == num) if (id == num) id = id ; else id = id ;



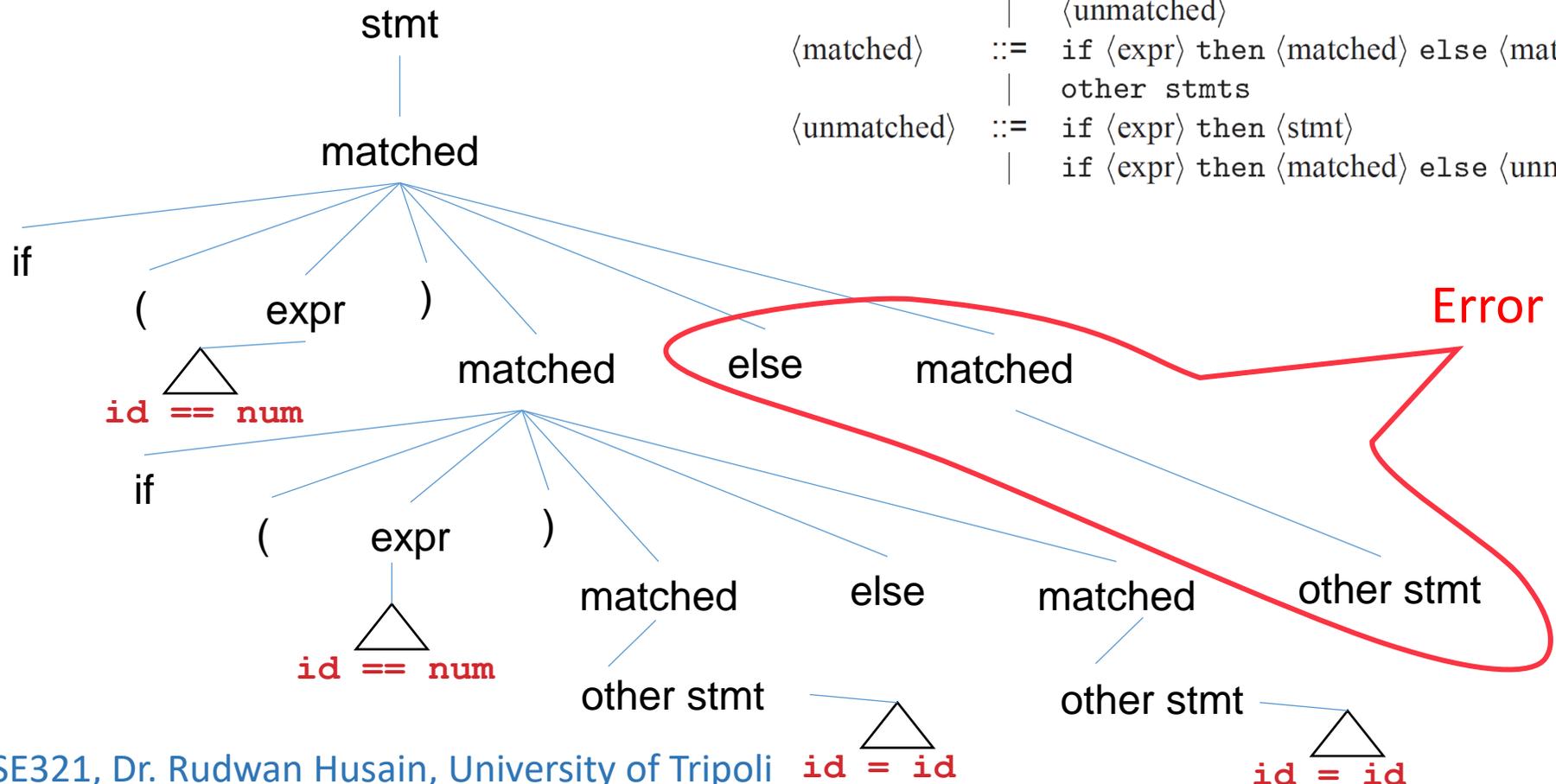
ELIMINATING AMBIGUITY

حذف الالتباس من جمل الاختبار if

جرب منتج matched كبداية لإعراب سلسلة البطاقات السابقة:

if (id == num) if (id == num) id = id ; else id = id ;

$\langle \text{stmt} \rangle$::=	$\langle \text{matched} \rangle$
		$\langle \text{unmatched} \rangle$
$\langle \text{matched} \rangle$::=	if $\langle \text{expr} \rangle$ then $\langle \text{matched} \rangle$ else $\langle \text{matched} \rangle$
		other stmts
$\langle \text{unmatched} \rangle$::=	if $\langle \text{expr} \rangle$ then $\langle \text{stmt} \rangle$
		if $\langle \text{expr} \rangle$ then $\langle \text{matched} \rangle$ else $\langle \text{unmatched} \rangle$



ELIMINATING AMBIGUITY

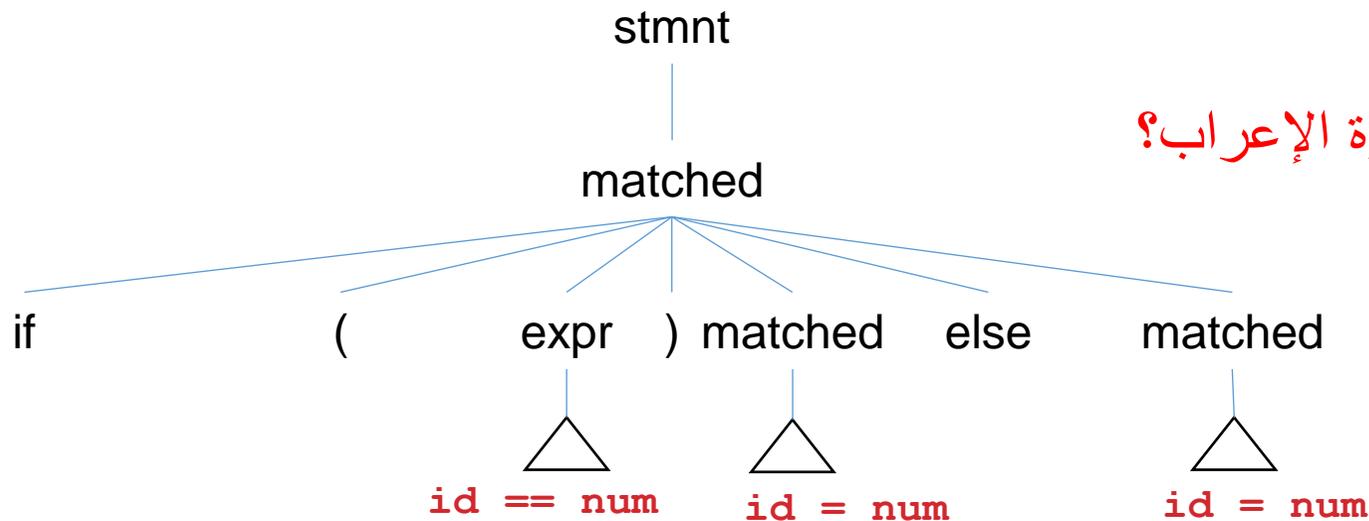
حذف الالتباس من جمل الاختبار if

مثال آخر: جزء من برنامج مصدري به التعابير الشرطية,

```
if (x==0)
    z = 0;
else
    z = 1;
```

$\langle \text{stmt} \rangle$	$::=$	$\langle \text{matched} \rangle$
		$\langle \text{unmatched} \rangle$
$\langle \text{matched} \rangle$	$::=$	if $\langle \text{expr} \rangle$ then $\langle \text{matched} \rangle$ else $\langle \text{matched} \rangle$
		other stmts
$\langle \text{unmatched} \rangle$	$::=$	if $\langle \text{expr} \rangle$ then $\langle \text{stmt} \rangle$
		if $\langle \text{expr} \rangle$ then $\langle \text{matched} \rangle$ else $\langle \text{unmatched} \rangle$

البطاقات tokens : if (id == num) id = num ; else id = num ;



موضوعنا التالي:

الإعراب

PARSING

Top-down parsing

Left recursion

Recursion [عامّة] استدعاء ذاتي
recursion [تقنية] تكرار
معاودة؛ نداء ذاتي [الحاسوب]