الاتومات واللغات الصورية

المحاضرة الرابعة — الاثنين 2017/10/23

L (تعبير منتظم) = { } ← اللغة التي يعبر عنها (يولدها) (يعرفها) التعبير المنتظم

مثال: $\Sigma = \{a, b, c\}$

$L(a) = \{a\}, L(b) = \{b\}, L(c) = \{c\}$

$L(a+b) = L(a) \cup L(b) = \{a\} \cup \{b\} = \{a, b\}$

* إذا كان $\{a, b, c\}$ لغة ما فإنه يمكننا كتابتها بالشكل:

$L(a+b+c) = L(a) \cup L(b) \cup L(c)$

$L(ab) = L(a) \cdot L(b)$ *

$= \{a\} \cdot \{b\} = \{ab\}$

$L(ab+cda) = L(ab) \cup L(cda)$ *

$= L(a) \cdot L(b) \cup L(c) \cdot L(d) \cdot L(a)$

$= \{a\} \cdot \{b\} \cup \{c\} \cdot \{d\} \cdot \{a\}$

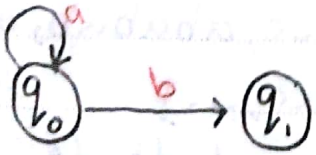
$= \{ab\} \cup \{cda\} = \{ab, cda\}$

التعبير المنتظم $ab+cda$ إما يولد اللغة ab أو اللغة cda

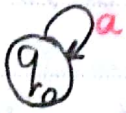
* التعبير المنتظم $a+ac+bda+db$

$L(a+ac+bda+db) = \{a, ac, bda, db\}$

b هو مدخل للانتقال من الحالة q_0 إلى الحالة q_1



التعبير المنتظم a^*b



التعبير المنتظم a^*

* $L(a^*) = \{\epsilon, a, aa, aaa, \ldots\}$
* $L((ab)^*) = \{\epsilon, ab, abab, ababab, \ldots\}$

Scanned by CamScanner

* $L(a^*b^*) = L(a^*).L(b^*) = \{\epsilon, a, aa, aaa, \ldots\}.\{\epsilon, b, bb, bbb, \ldots\}$
$= \{\epsilon, b, bb, bbb, \ldots, a, ab, abb, abbb, \ldots, aa, aab, aabb, aabbb, \ldots\}$

$L(a^*b^*)$ اللغة التي يولدها التعبير المنتظم a^*b^* هي إما ϵ أو مجموعة من السلاسل المشكلة من تعاقب لـ a مرة واحدة أو أكثر أو مجموعة السلاسل المشكلة من تعاقب لـ b مرة واحدة أو أكثر أو مجموعة السلاسل التي تبدأ من أي تكرار لـ a يعقبها أي تكرار لـ b

* السلاسل تتألف من تعاقب $L(a^+) = \{a, aa, aaa, \ldots\}$ لـ a مرة واحدة على الأقل

* $L(ab + c^*) = \{ab, \epsilon, c, cc, ccc, \ldots\}$
* $L(a^* + b^*) = \{\epsilon, a, aa, aaa, \ldots, b, bb, bbb, \ldots\}$

تتألف من ϵ أو السلاسل المشكلة من تعاقب لـ a مرة واحدة على الأقل أو السلاسل المشكلة من تعاقب لـ b مرة واحدة على الأقل.

* $L(a+c)^* = \{\epsilon, a, c, aa, cc, ac, ca, aaa, ccc, acc, caa, cac, cca, aca, aac, \ldots\}$

(تكرار مرة واحدة — ثلاث تكرارات — تكرارين)

$L(a+c)^* = (L(a) \cup L(c))^* = (\{a\} \cup \{c\})^* = \{a, c\}^*$
$= \{\epsilon, a, c, aa, ac, ca, cc, aaa, aac, aca, acc, \ldots\}$

$\{ac\}^2 = \{a, c\}.\{a, c\} = \{aa, ac, ca, cc\}$

$\{a, c\}^3 = \{a.c\}^2.\{a, c\} = \{a, c\}.\{a, c\}^2 = \{a, c\}.\{aa, ac, ca, cc\}$
$= \{aaa, aac, aca, acc, caa, cac, cca, ccc\}$

وهكذا ...

* $L(ab+c)^* = \{\epsilon, ab, c, abab, abc, cab, cc, ababab, ababc, abcab, abcc, cabab, cabc, ccab, ccc, \ldots\}$
* $L(ab^*c + d^*) = \{ac, abc, ab^2c, ab^3c, \ldots, \epsilon, d, d^2, d^3, \ldots\}$

$ab^*c = \{ac, abc, ab^2c, \ldots\}$

$\{\epsilon, b, b^2, b^3, \ldots\}$

$dabc \notin L(ab^*c + d^*)$

* $\Sigma = \{a, b\}$ $L(a+b)^* = \Sigma^*$ كل السلاسل الممكن توليدها من Σ
* $L(a^* + b^*) = \{\epsilon, a, a^2, a^3, \ldots, b, b^2, b^3, \ldots\}$

Scanned by CamScanner

* $(a+b)^* \neq a^* + b^*$

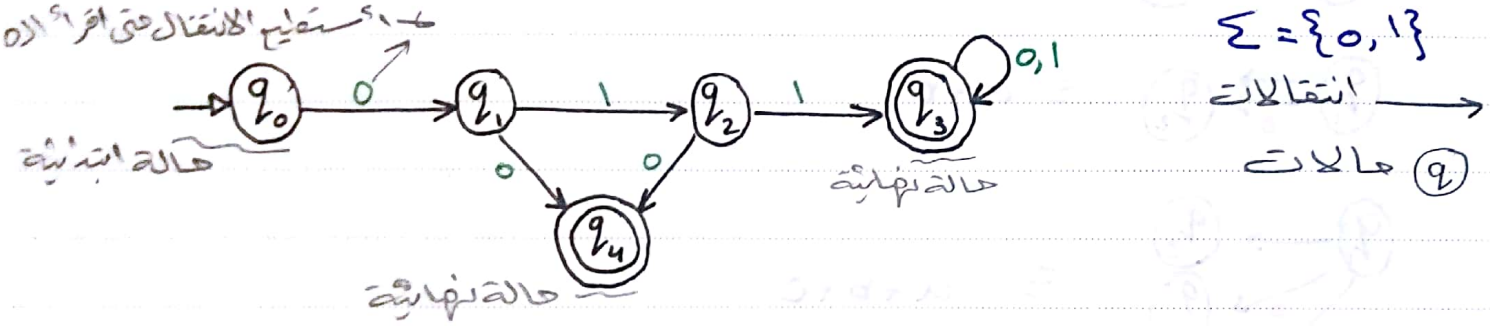
$L(a+b+c)^* = \Sigma^*$ $\quad \Sigma = \{a, b, c\}$ *

* $L(a^* + b^* + c^*) = \{\epsilon, a, a^2, a^3, \ldots, b, b^2, b^3, \ldots, c, c^2, c^3, \ldots\}$

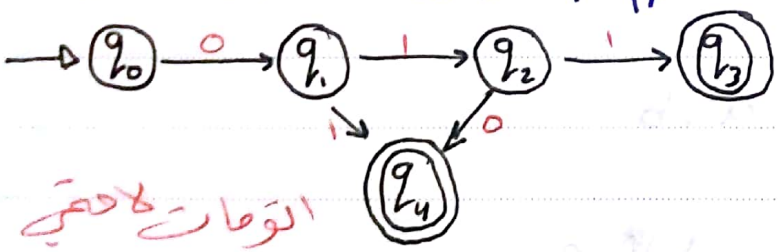
* $L(a^* . b^*)$

.	ϵ	b	b^2	b^3	.....
ϵ	ϵ	b	b^2	b^3	....
a	a	ab	ab^2	ab^3	....
a^2	a^2	a^2b	a^2b^2	a^2b^3	...
a^3	a^3	a^3b	a^3b^2	a^3b^3	...
⋮	⋮	⋮	⋮	⋮	...

الاتومات المنتهي الحتمي : آلة للتعرف على كلمات تنتمي إلى لغة ذات صفة معينة

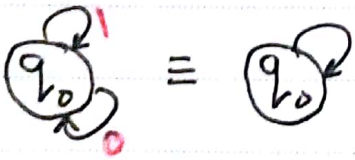


اتومات حتمي (يخرج من كل حالة سهم بعدد رموز اللغة ولا ولكل رمز انتقال واحد من كل حالة)

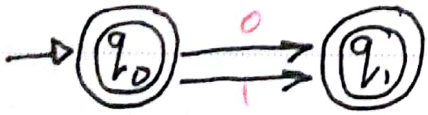


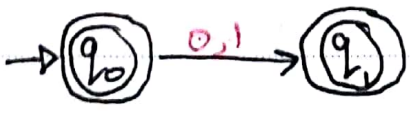
اتومات لاحتمي

لكل اتومات حالة ابتدائية واحدة فقط و الحالة النهائية يمكن أن تكون أكثر من واحدة .



$\equiv$ $\equiv (0+1)^*$ التعبير المنتظم



$\equiv$  $\equiv 0+1$ التعبير المنتظم

انتهت المحاضرة

Scanned by CamScanner