مقرر اللغات الصورية

د. هلا نصار

محاضرات الأسبوع 1

الفصل الصيفي 2022-2023

اللغات الصورية Formal language: هي اللغات التي يمكن تعريفها بواسطة عدد من القواعد.

الرمز symbol: هو غير قابل للتجزئة، مثل الحروف الإنكليزية، العربية، الأرقام..

فمثلاً hc ليس رمز لأنه قابل للتجزئة إلى h و c.

الأبجدية alphabet: هي مجموعة منتهية وغير خالية من الرموز وتتميز بأنه لا يمكن توليد أي رمز منها بواسطة بقية الرموز ويرمز عادة للأبجدية بالرمز $\sum$.

إن $\emptyset = \{\}$ ليست أبجدية لأنها مجموعة خالية

مجموعة الأعداد الطبيعية N ليست أبجدية لأنها مجموعة غير منتهية

المجموعة $\sum=\{A,B,AB\}$ لا تشكل أبجدية لأنها مجموعة تحوي عنصر قابل للتجزئة وبالتالي ليس رمز.

من المجموعات التي هي أبجدية

$\sum=\{a,b,c\}, \sum=\{0,1\}, \sum=\{0,1,\ldots,9\}$

السلسلة string: هي تعاقب أي عدد منتهي من الرموز المأخوذ من أبجدية ما دون وجود فراغات بينها. ويمكن للسلسلة أن تتألف من رمز واحد فقط.

من أجل الأبجدية $\sum=\{0,1\}$ عندئذ يكون كل منها 001101110، 101110، 011 سلاسل من الأبجدية $\sum$.

عادة يرمز للسلسلة برموز صغيرة x,y,z,w,r,v,u مثلا لدينا السلسلة $x=ab, z=01101$

طول السلسلة length of string هو عدد الرموز في السلسلة ويرمز لطول السلسلة w بالرمز $|w|$

إذا كان لدينا السلسلة $v=aaababb$ من الأبجدية $\sum=\{a,b\}$ عندئذ يكون طول السلسلة $|v|=7$

إذا كان لدينا السلسلة $x=34473$ من أبجدية النظام العشري عندئذ طول السلسلة x يعطى $|x|=5$

السلسلة الفارغة empty string هي سلسلة لا تحوي أي رموز وطولها يساوي الصفر ونرمز لها بالرمز ε

حيث يكون $|\varepsilon|=0$ ويجب الانتباه أن السلسلة الفارغة هي معرفة على أي أبجدية

العمليات على السلاسل

التعاقب على السلاسل :concatenation

إن تعاقب سلسلتين x,y هو سلسلة مشكلة من توضع رموز السلسلة الأولى x متبوعة مباشرة بالسلسلة الثانية y ونرمز لعملية التعاقب بالرمز (.)

لتكن لدينا الابجدية $\sum=\{s,t\}$ ولتكن لدينا السلسلتين x=sssts, y=tst عندئذ يكون :

$x.y=ssststst$

$y.x=tstsssts$

ومنه نلاحظ أن تعاقب السلاسل عملية غير تبديلية وهي عملية تجميعية

السلسلة الفارغة ε هي العنصر الحيادي بالنسبة لعملية التعاقب

$ssssts.\ \varepsilon = \varepsilon\ .\ sssts = sssts$ أي $x\ \varepsilon\ = \varepsilon\ x=x$

بادئة سلسلة prefix

تكون السلسلة x هي بادئة السلسلة y إذا وجدت السلسلة v تحقق y=xv

لتكن لدينا السلسلة v=abbaaba عندئذ تكون مجموعة كل البادئات لهذه السلسلة هي:

{Ɛ,a,ab,abb,abba,abbaa,abbaab,abbaaba}

أي اول عنصر في مجموعة البادئات هي السلسلة الفارغة، ثم الرمز الأول، الرمز الأول والثاني من السلسلة، الرمز الأول والثاني والثالث، وهكذا مع المحافظة على ترتيب الرموز في السلسلة

لاحقة سلسلة suffix

تكون السلسلة x هي لاحقة السلسلة y إذا وجدت السلسلة v تحقق y=vx

لتكون لدينا السلسلة z=ababba عندئذ تكون مجموعة كل اللاحقات لهذه السلسلة هي:

{Ɛ,a,ba,bba,abba,babba,ababba}

أي أول عنصر في مجموعة اللاحقات دوماً يكون السلسلة الفارغة ثم يليها الرمز الأخير ومن ثم الأخير والذي يسبقه وهكذا مع المحافظة على ترتيب الرموز.

السلسلة الجزئية sub string:

تكون السلسلة x سلسلة جزئية من السلسلة y إذا كانت كل رموز السلسلة x موجودة في السلسلة y وبنفس الترتيب.

لتكن لدينا السلسلة v= 1110010 ومنه تكون كل السلاسل التالية هي سلاسل جزئية منها 11, 100,010, 1001

ومنه نجد أن البادئات واللاحقات هي سلاسل جزئية.

قوة سلسلة power of string:

قوة سلسلة x من الدرجة n هي عبارة عن تعاقب هذه السلسلة n مرة.

$x^n = x.x....x$: n مرة

ليكن لدينا $x=bba=b^2a$ ومنه يكون لدينا: $x^3=x.x^2=x^2.x=x.x.x= bbabbabba=b^2ab^2ab^2a$

ليكن لدينا $z=ab^3a^2$ احسب z^3

قوة أبجدية power of alphabet:

إن قوة أبجدية من الدرجة n هي مجموعة من السلاسل المولدة من الابجدية $\sum$ ذات الطول n ويرمز لها بالشكل $\sum^n$

لتكن لدينا الابجدية $\sum=\{a,b\}$ عندئذ يكون لدينا:

$\sum^0=\{\varepsilon\}$

$\sum^1=\{a,b\}$

$\sum^2=\sum.\sum=\{a,b\}.\{a,b\}=\{aa,ab,ba,bb\}$

$\sum^3=\sum^2.\sum=\sum.\sum^2=\sum.\sum.\sum=\{a,b\}.\{a,b\},\{a,b\}=\{a,b\}\{aa,ab,ba,bb\}=\{aaa,aab,aba,abb,baa,bab,bba,bbb\}$

ومنه نعرف $\sum^*$ على انها مجموعة كل السلاسل الممكن توليدها من رموز الأبجدية $\sum$ وهي مجموعة غير منتهية أي:

$\sum^*=\sum^0 \cup \sum^1 \cup ...$

ومنه تكون $\sum^*$ المشكلة من الأبجدية $\{a,b\}$ هي مجموعة غير منتهية من كل السلاسل التي يمكن تشكيلها من رموز الأبجدية a,b

يرمز للمجموعة غير المنتهية المكونة من كل السلاسل الممكن توليدها من أبجدية ما عدا السلسلة الفارغة بالرمز $\sum^+$ ومنه يكون لدينا :

$$\sum^+ = \sum^* - \sum^0 = \sum^* - \{\varepsilon\}$$

$$\sum^+ \cup \{\varepsilon\} = \sum^*$$

$$\{\varepsilon\} = \sum^* - \sum^+$$

اللغة languages: هي مجموعة السلاسل المختارة من المجموعة $\sum^*$ والمولدة ممن الابجدية $\sum$ ويرمز لها بالرمز L وهي مجموعة محتواه في $\sum^*$

ومنه ليس بالضرورة أن تحوي اللغة كل السلاسل المولدة من أبجدية ما. وعادة اللغة قد تكون منتهية وقد لا تكون منتهية.

ليكن لدينا الأبجدية $\sum=\{A,B\}$ عندئذ يمكن توليد اللغة المكونة من كل السلاسل التي تحوي تعاقب AA عندئذ نجد أن اللغة غير منتهية لوجود عدد لا نهائي من السلاسل التي يمكن توليدها من الأبجدية وتحوي AA

ليكن لدينا الابجدية $\sum=\{a,b,c\}$ المطلوب:

ما هي اللغة L1 المولدة من الابجدية والتي تحوي الرمز b مرتين متتاليتين على الأقل؟

ما هي اللغة L2 المولدة من الابجدية والتي تحوي الرمز b مرتين فقط؟ لاحظ أنها لغة محتواه في L1

ما هي اللغة L3 المولدة من الأبجدية والتي تبدأ بالرمز C؟ لاحظ أن السلسلة الفارغة لا تنتمي إلى اللغة.

العمليات على اللغات set operations

اتحاد لغتين union

اتحاد لغتين L1,L2 يمثل بالشكل التالي L1 U L2 وهو عبارة عن اللغة التي تحوي جميع السلاسل الموجودة في L1 أو L2 أو كلاهما

$$L_1 \cup L_2 = \{x;\ x \in L_1\ or\ \ x \in L_2\}$$

تقاطع لغتين intersection

تقاطع لغتين L1,L2 يمثل بالشكل التالي L1 ∩ L2 وهو عبارة عن اللغة التي تحوي جميع السلاسل الموجودة في L1 و L2 معاً

$$L_1 \cap L_2 = \{x;\ x \in L_1\ and\ \ x \in L_2\}$$

فرق لغتين difference

فرق لغتين L1,L2 يمثل بالشكل التالي L1 - L2 وهو عبارة عن اللغة التي تحوي السلاسل الموجودة في L1 وغير الموجودة في L2

$$L_1 - L_2 = \{x;\ x \in L_1\ and\ \ x \notin L_2\}$$

متمم لغة complement

متمم اللغة L المعرفة على أبجدية $\sum$ هي اللغة التي تحوي السلاسل الموجودة في $\sum^*$ وغير الموجودة في L

$$\bar{L} = \{x;\ x \in \Sigma^*\ and\ \ x \notin L\} = \Sigma^* - L$$

تعاقب لغتين concatenation

تعاقب لغتين L1,L2 يمثل بالشكل التالي L1 . L2 وهو عبارة عن اللغة المكونة من السلاسل المشكلة من تعاقب سلاسل L1 متبوعة بسلاسل اللغة L2

$$L_1.L_2 = \{xy \, ; x \in L_1 \, , y \in L_2\}$$

اغلاق لغة kleen star, closure:

اغلاق اللغة L المعرفة على أبجدية $\sum$ نرمز لها بالرمز *L وهي اللغة المشكلة من مجموعة كل السلاسل الناتجة عن تعاقب السلاسل في L

$$L^* = L^0 \cup L^1 \cup L^2 \cup \dots\dots$$

$$L^+ = L^* - \{\varepsilon\}$$
$$L^0 = \{\varepsilon\}, L^1 = L$$
$$L^2 = L.L \, , L^3 = L^2.L = L.L^2 = L.L.L \quad , \quad \dots\dots$$

مثال: لتكن لدينا اللغتين L_1 و L_2 المولدتين من الأبجدية $\Sigma = \{a, b\}$ حيث:

$$L_1 = \{a, ab, aaaa\}$$
$$L_2 = \{bb, ab\}$$

فيكون التقاطع:

$$L_1 \cap L_2 = \{ab\}$$

والاتحاد:

$$L_1 \cup L_2 = \{a, ab, aaaa, bb\}$$

والفرق:

$$L_1 - L_2 = \{a, aaaa\}$$
$$L_2 - L_1 = \{bb\}$$

ومتمم L_1 :

$$\overline{L_1} = \Sigma^* - \{a, ab, aaaa\} = \{\varepsilon, aa, aaa, bbb, abb, \dots\}$$

ومتمم L_2 :

$$\overline{L_2} = \Sigma^* - \{bb, ab\} = \{\varepsilon, a, aa, aba, \dots\}$$

تعاقب $L_1 . L_2$:

$$L_1 . L_2 = \{abb, aab, abbb, abab, aaaabb, aaaaab\}$$

تعاقب $L_2 . L_1$:

$$L_2 . L_1 = \{bba, bbab, bbaaaa, aba, abab, abaaaa\}$$

إغلاق L_2 (أي إيجاد L_2^*) :

$$L_2^0 = \{\varepsilon\}, \qquad L_2^1 = L_2 = \{bb, ab\}$$
$$L_2^2 = L_2 . L_2 = \{bb, ab\}\{bb, ab\} = \{bbbb, bbab, abbb, abab\}$$
$$L_2^3 = L_2 . L_2^2 = \{bb, ab\}\{bbbb, bbab, abbb, abab\}$$
$$L_2^3 = \{b^6, b^4ab, b^2ab^3, b^2abab, ab^5, ab^3ab, abab^3, ababab\}$$
$$\vdots$$
$$L_2^* = \{\varepsilon, bb, ab, bbbb, bbab, abbb, \dots \dots\}$$
$$L_2^* = L_2^0 \cup L_2^1 \cup L_2^2 \cup \dots \dots$$

ملاحظة: السلسلة $ababab$ يمكن كتابتها بالشكل $(ab)^3$

ونلاحظ أن : $(ab)^3 \neq a^3b^3$

خاصة: $\phi^0 = \phi^* = \{\varepsilon\}$

اللغات المنتظمة

يمكن التعبير عن اللغة المنتظمة من خلال:

- تعبير منتظم
- أوتومات منتهي
- قواعد منتظمة

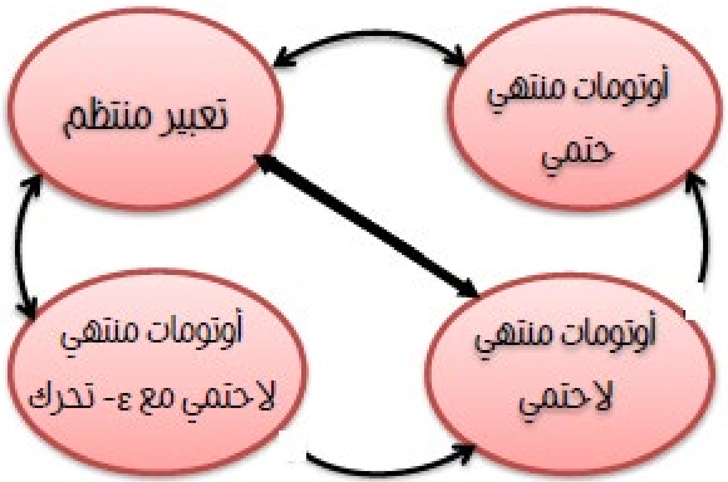
وهناك عدة أنواع من الأوتومات المنتهي جميعها متكافئة:

أوتومات منتهي حتمي

أوتومات منتهي لاحتمي

أوتومات منتهي لاحتمي مع ع-تحرك

حيث نستطيع تحويل كل اوتومات منتهي لاحتمي مع ع-تحرك إلى اوتومات منتهي لاحتمي والعكس ونستطيع تحويل كل أوتومات منتهي لاحتمي إلى أوتومات منتهي حتمي والعكس. وكل أوتومات منتهي يقبل لغة منتظمة نعبر عنها بواسطة التعابير المنتظمة.



التعابير المنتظمة *Regular Expressions*

تستخدم التعابير المنتظمة في تعريف وتحديد شكل اللغات المنتظمة.

حيث يمكن تعريف اللغة المنتظمة على أبجدية كما يلي:

- {ε} هي لغة منتظمة على $\sum$.
- إذا كان a حرف من حروف الأبجدية عندئذٍ تكون المجموعة {a} لغة منتظمة على $\sum$.
- إذا كانت L لغة منتظمة على $\sum$ فإن L^*, L^n هي لغات منتظمة على $\sum$.
- إذا كانت كل من L1,L2 لغات منتظمة فإن $L1.L2\ ,\ L1 \cap L2,\ L1 \cup L2, L1-L2$ لغات منتظمة على $\sum$
- لا يوجد لغات منتظمة أخرى.

لتكن $\sum$ أبجدية من الرموز عندئذٍ:
Ø هو التعبير المنتظم الذي يحدد اللغة الفارغة.

$$\underbrace{L(\phi)}_{\text{اللغة التي يعرفها التعبير المنتظم } \phi} = \phi = \{\ \}$$

ε هو التعبير المنتظم الذي يحدد اللغة التي تحوي السلسلة الفارغة:

$$L(\varepsilon) = \{\varepsilon\}$$

من أجل أي رمزه a من الأبجدية $\sum$ فإن a هو تعبير منتظم يحدد اللغة:

$$L(a) = \{a\}$$

إذا كان E,F تعبيران منتظمان فإن التعبير E+F هو تعبير منتظم يحدد اجتماع اللغتين L(E), L(F):

$$L(E + F) = L(E) \cup L(F)$$

إذا كان E,F تعبيران منتظمان فإن التعبير E.F هو تعبير منتظم يحدد تعاقب اللغتين L(E), L(F) :

$$L(EF) = L(E)L(F)$$

إذا كان E تعبير منتظم فإن *E هو تعبير منتظم (تكرار لـ E من الصفر إلى تكرارات غير منتهية) واللغة التي يحددها:

$$L(E^*) = (L(E))^*$$

إذا كان E تعبير منتظم فإن (E) هو تعبير منتظم ويحدد نفس اللغة التي يحددها التعبير المنتظم E:

$$L((E)) = L(E)$$

إذا كانت الابجدية $\Sigma = \{a, b, c\}$ ، عندئذٍ :

❶ تكون اللغة المولدة بالتعبير المنتظم $a + b + c$ هي :

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

❷ تكون اللغة المولدة بالتعبير المنتظم $ab + cda$ هي:

$$L(ab + cda) = L(ab) \cup L(cda)$$
$$= L(a)L(b) \cup L(c)L(d)L(a)$$
$$= \{a\}\{b\} \cup \{c\}\{d\}\{a\}$$
$$= \{ab\} \cup \{cda\}$$

❸ ليكن لدينا التعبير المنتظم: $a + ac + bda + db$ فتكون سلاسل اللغة التي يولدها تكون إما a أو ac أو bda أو db :

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

إذا كان لدينا التعبير المنتظم *(ab) فإن اللغة التي يولدها هذا التعبير هي عناصرها إما ε أو السلاسل المكونة من تعاقب (ab) إما مرة واحدة أو أكثر من مرة

$$L((ab)^*) = \{\varepsilon, ab, abab, \dots \dots\}$$

أما في حالة كان التعبير المنتظم a*b* فإن اللغة التي يولدها هي:

$$L(a^*b^*) = L(a^*)L(b^*) = \{\varepsilon, a, aa, \dots\}\{\varepsilon, b, bb, \dots\}$$
$$= \{\varepsilon, a, aa, \dots, b, bb, \dots, ab, abb, \dots, aab, \dots\}$$

وهي اللغة التي عناصرها إما ε أو مجموعة السلاسل المشكلة من تعاقب a مرة واحدة أو أكثر أو مجموعة السلاسل المشكلة من تعاقب b مرة واحدة أو أكثر أو مجموعة السلاسل التي تبدأ من أي تكرار a يعقبها أي تكرار b وهذا التكرار يكون من الصفر إلى اللانهاية.
يمكن إيجاد اللغة التي يولدها هذا التعبير المنتظم عن طريق الجدول التالي:

$a^* \backslash b^*$	ε	b	bb	bbb	$\dots$
ε	ε	b	bb	bbb	$\dots$
a	a	ab	abb	$abbb$	$\dots$
aa	aa	aab	$aabb$	$aabbb$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

هل السلسلة abab تنتمي الى اللغة L(a*b*)؟ لا تنتمي لأنها ليست تعاقب لأي تكرار للرمز a يليه أي تعاقب للرمز b

$$L(a+c)^* = \left(L(a+c)\right)^* = \left(L(a) \cup L(c)\right)^*$$
$$= (\{a\} \cup \{c\})^* = \{a,c\}^* = \underbrace{\{a,c\}^0}_{\{\varepsilon\}} \cup \underbrace{\{a,c\}^1}_{\{a,c\}} \cup \underbrace{\{a,c\}^2}_{\{aa,ac,ca,cc\}} \cup \dots$$
$$\Rightarrow L(a+c)^* = \{\varepsilon, a, c, aa, ac, ca, cc, \dots\}$$

$$L(\boldsymbol{a^*}) = \{\varepsilon, a, aa, aaa, aaaa, \dots\dots\}$$
$$L(\boldsymbol{ab+c^*}) = L(ab) \cup \{\varepsilon, c, cc, ccc, \dots\} = \{ab, \varepsilon, c, cc, ccc, \dots\}$$
$$L(\boldsymbol{a^*+b^*}) = \{\varepsilon, a, aa, \dots, b, bb, bbb, \dots\}$$

$$L(ab+c)^* = \left(L(ab) \cup L(c)\right)^* = \{\varepsilon, ab, c, abab, abc, cab, cc, \dots\}$$
$$L(\boldsymbol{ab^*c+d^*}) = L(ab^*c) \cup L(d^*) = \{ac, abc, abbc, \dots\} \cup \{\varepsilon, d, dd, \dots\}$$
$$= \{ac, abc, abbc, abbbc, \dots, \varepsilon, d, dd, \dots\}$$

هل dabc ينتمي إلى اللغة السابقة؟ لا ينتمي لأن اللغة المولدة بالتعبير هي إما السلسلة الفارغة أو تعاقب d مرة أو أكثر أو تعاقب a مع أي تكرار للرمز b يليه c.

في حال كانت اللغة هي *L(ab*c+d) فتكون dabc تنتمي.

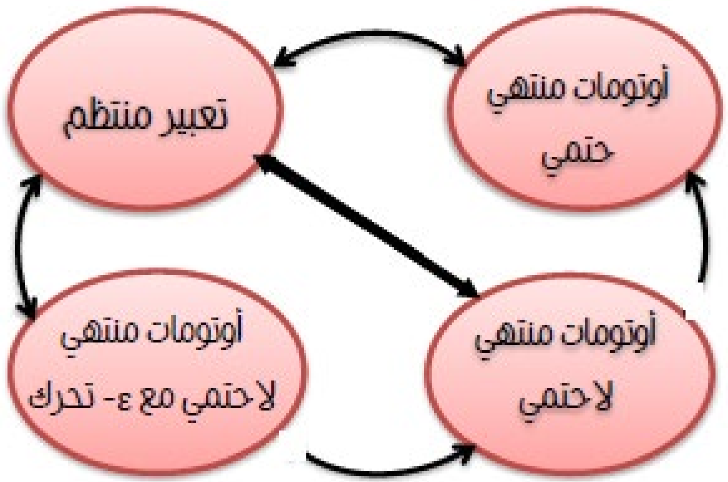
$$L(ab^*c+d)^* = \left(L(ab^*c) \cup L(d)\right)^* = (\{ac, abc, ..\} \cup \{d\})^*$$
$$= (\{d, ac, abc, abbc, \dots\})^* = \{\varepsilon, ac, abc, abbc, \dots, dabc,, abcd, \dots\}$$

لتكن لدينا الأبجدية $\sum=\{a,b\}$ عندئذ $L(a+b)^*=\sum^*$ وهي كل السلاسل الممكن توليدها من $\sum$

أما $L(a^*+b^*)$ هي كل السلاسل المولدة من ε أو تعاقب للرمز a مرة أو أكثر أو تعاقب للرمز b مرة أو أكثر.

ومنه $L(a+b)^* \neq L(a^*+b^*)$

اللغات المنتظمة



يمكن التعبير عن اللغة المنتظمة من خلال:

- تعبير منتظم
- أوتومات منتهي
- قواعد منتظمة

وهناك عدة أنواع من الأوتومات المنتهي جميعها متكافئة:

أوتومات منتهي حتمي

أوتومات منتهي لاحتمي

أوتومات منتهي لاحتمي مع ε-تحرك

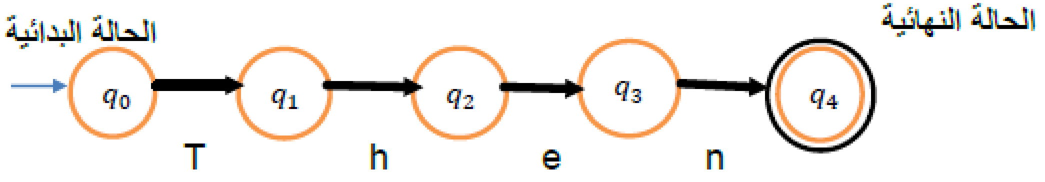
حيث نستطيع تحويل كل اوتومات منتهي لاحتمي مع ε-تحرك إلى اوتومات منتهي لاحتمي والعكس ونستطيع تحويل كل أوتومات منتهي لاحتمي إلى أوتومات منتهي حتمي والعكس. وكل أوتومات منتهي يقبل لغة منتظمة نعبر عنها بواسطة التعابير المنتظمة.

الأوتومات المنتهي Finite Automate

الاوتومات المنتهي هو نموذج رياضي على شكل بيان Graph أو جدولTable.

- يحتوي على عدد منتهي من الحالات عادة ما تسمى qi حيث i=0..n أي عدد الحالات n+1،
- يرمز للحالات بالدائرة والحالة الابتدائية يوضع أمامها سهم والحالات النهائية دائرتين.
- الانتقالات بين الحالات توضع على الأسهم الواصلة بين كل حالة وأخرى.
- بكل أوتومات يوجد حالة ابتدائية واحدة وقد يكون هناك أكثر من حالة نهائية.

للتعرف على كلمة then نجد أن الاوتومات المنتهي الحتمي المكافئ

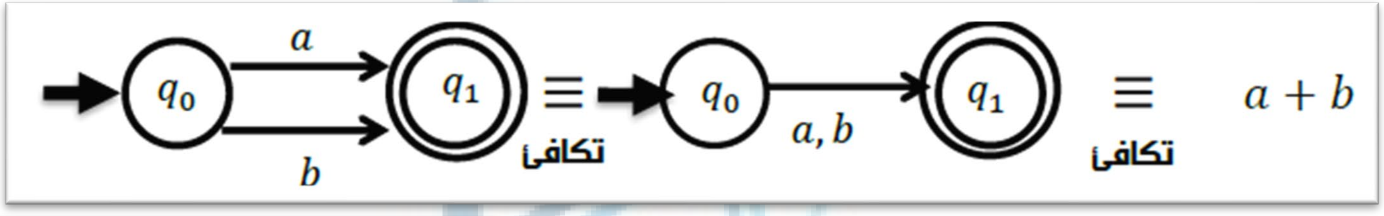


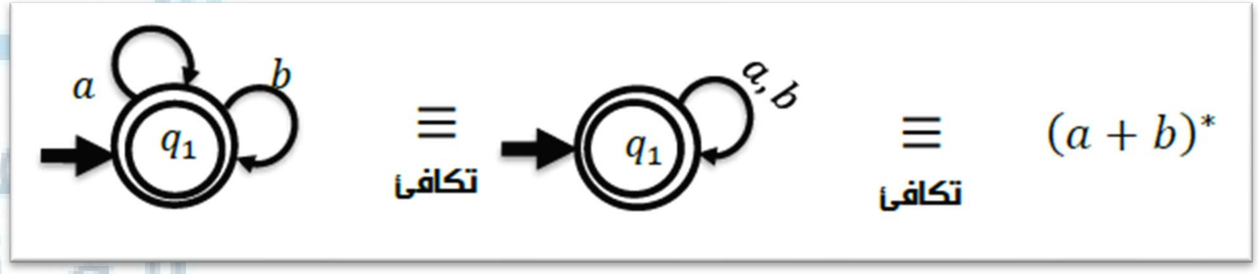
q_i: هي اسماء الحالات.

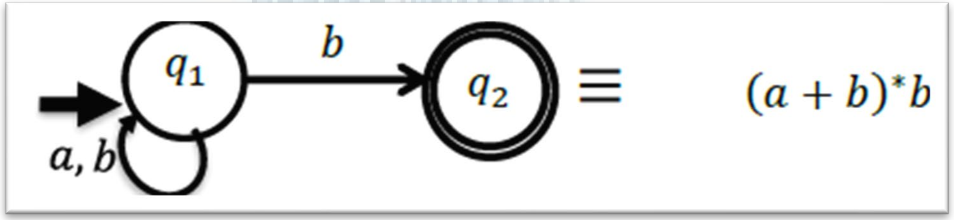
سبب دراسة الأوتومات :

- يستخدم في تصميم الدارات الالكترونية وبروتوكولات الاتصال
- يستخدم في برامج معالجة النصوص للبحث عن الكلمات في ملف أو عبر الويب.
- تلعب دوراً هاماً في صناعة المترجمات compilers على مستوى المفردات كمحلل لفظي lexical analyzes
- تكون وظيفة الأوتومات التعرف على لغة ما.

كتابة التعبير المنتظم من خلال أوتومات منتهي







الأوتومات المنتهي الحتمي

deterministic finit automata

الأوتومات المنتهي الحتمي هو أوتومات عدد حالاته منتهي وكل حالة qi يخرج منها عدد أسهم بعدد رموز الأبجدية وكل سهم يكون مختلف عن الآخر بالرمز المختار له.

نقول عن سلسلة ما أنها مقبولة في الأوتومات المنتهي الحتمي إذا وصل الأوتومات إلى حالة نهائية بعد قراءة السلسلة بأكملها حيث نبدأ بالقراءة من الحالة الابتدائية للأوتومات.

يرمز لها اختصاراً DFA - deterministic finit automata ويعرف الأوتومات المنتهي بالخماسية:

$M=(Q,\sum,\delta,q0,F)$

حيث

Q مجموعة منتهية من الحالات للأوتومات وهي غير خالية

$\sum$ أبجدية الدخل

δ هو تابع الانتقال ويعرف بالشكل $\delta:Q \times \sum \rightarrow Q$ أي أن

$$\delta(q,x) = q' \, ; q,q' \in Q \, , x \in \Sigma$$

q0 الحالة الابتدائية وهي وحيدة دوماً وتنتمي لمجموعة الحالات Q

F هي مجموعة حالات النهائية وتكون محتواه في Q

الحالة الميتة: هي الحالة التي لا يمكن الانتقال منها إلى أحد الحالات النهائية.

الأوتومات التالي منتهي حتمي يعرف بالخماسية:

$M=(Q,\sum,\delta,q0,F)$

Q هي مجموعة {q0,q1,q2}

$\sum$ أبجدية الدخل {a,b}

δ هو تابع الانتقال ويعرف بالشكل $\delta:Q\ X \sum \rightarrow Q$ يعرف بالجدول التالي:

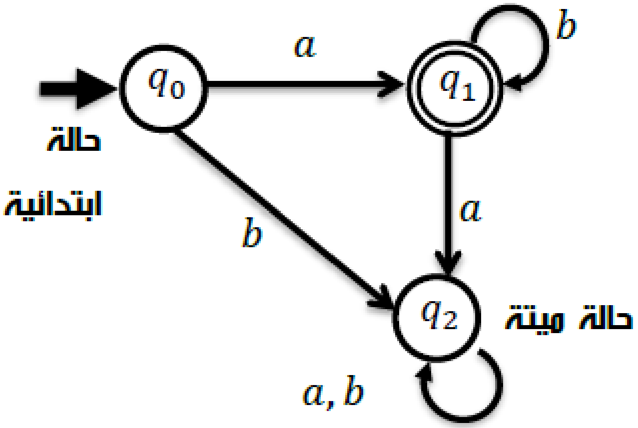
δ	a	b
q_0	q_1	q_2
q_1	q_2	q_1
q_2	q_2	q_2

q0 الحالة الابتدائية تنتمي لمجموعة الحالات Q

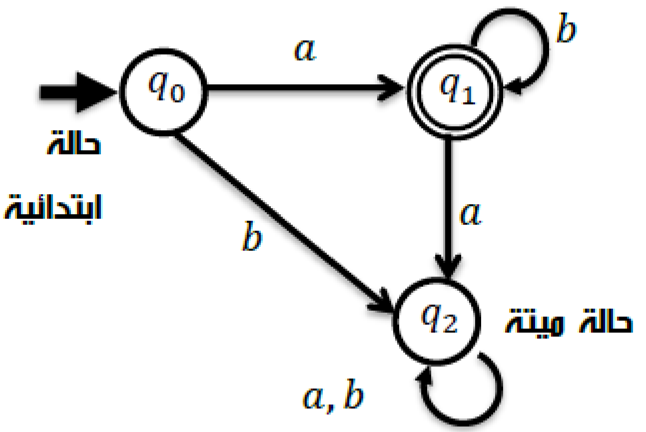
F هي مجموعة تتألف من حالة وحيدة {q1}

الحالة q2 هي حالة ميتة

ارسم الأوتومات المكافئ



هل يقبل الأوتومات السلاسل التالية abbb, aaaba, baaab, abba



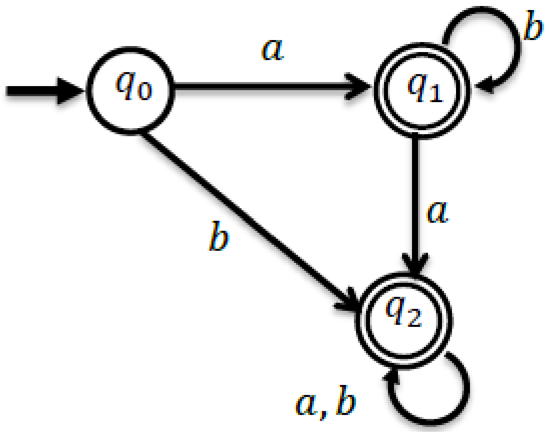
من أجل abbb يمكن اثبات إن كانت تنتمي أم لا من خلال إحدى الطريقتين:

$$q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_1 \xrightarrow{b} q_1 \xrightarrow{b} q_1$$

$$\delta(q_0, abbb) = \delta(\delta(q_0, a), bbb) = \delta(q_1, bbb) = \delta(\delta(q_1, b), bb)$$
$$\delta(q_1, bb) = \delta(\delta(q_1, b), b) = \delta(q_1, b) = q_1$$

وبما أن الحالة التي تم الوصول إليها عند نهاية السلسلة هي حالة نهائية فالسلسلة مقبولة في الاوتومات

نلاحظ أن التعبير المنتظم للأوتومات هو *ab



هل الاوتومات التالي منتهي حتمي؟

ما هو التعبير المنتظم للأوتومات؟

عند تعقب الانتقال بين الحالات وصولاً إلى الحالات النهائية انطلاقاً من الحالة الابتدائية يمكن ملاحظة وجود ثلاث خيارات لسلاسل مقبولة ومنه يمكن كتابة التعبير المنتظم للأوتومات هو

$$\underbrace{ab^*}_{\text{الطريق الأول}} \underbrace{+}_{\text{أو}} \underbrace{ab^*a(a+b)^*}_{\text{الطريق الثاني}} \underbrace{+}_{\text{أو}} \underbrace{b(a+b)^*}_{\text{الطريق الثالث}}$$

ليكن لدينا الأوتومات المنتهي الحتمي يعرف بالخماسية:

$M=(Q,\sum,\delta,q0,F)$

حيث

Q هي مجموعة $\{q0,q1,q2\}$

$\sum$ أبجدية الدخل $\{a,b\}$

δ هو تابع الانتقال ويعرف بالشكل $\delta{:}Q \times \sum \longrightarrow Q$ يعرف بالجدول التالي:

δ	a	b
q_0	q_0	q_1
q_1	q_2	q_2
q_2	q_2	q_2

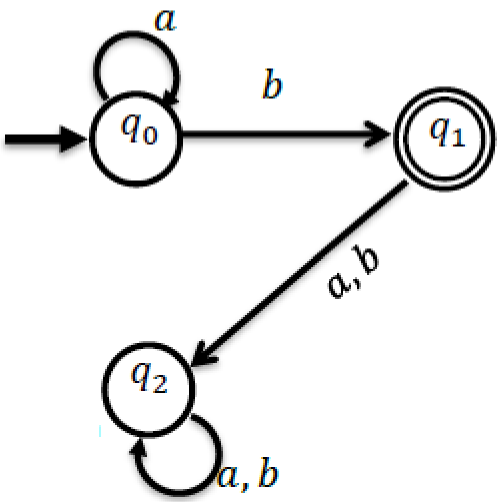
أو ما يكافئها بالشكل التالي

$$\delta(q_0,a)=q_0, \delta(q_0,b)=q_1\ , \delta(q_1,a)=q_2\ , \delta(q_1,b)=q_2$$
$$\delta(q_2,a)=q_2\ ,\ \ \delta(q_2,b)=q_2$$

q0 الحالة الابتدائية تنتمي لمجموعة الحالات Q

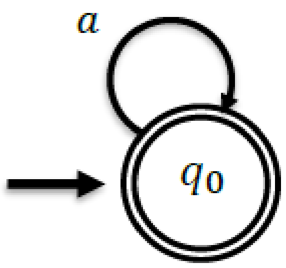
F هي مجموعة تتألف من حالة وحيدة $\{q1\}$

الحالة q2 هي حالة ميتة

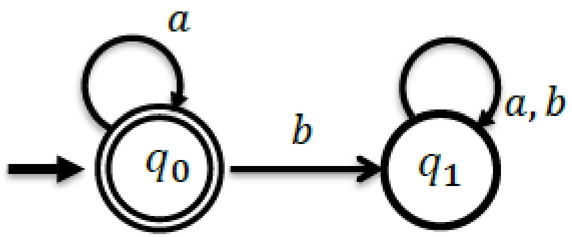


اللغة المنتظمة للأوتومات: a*b

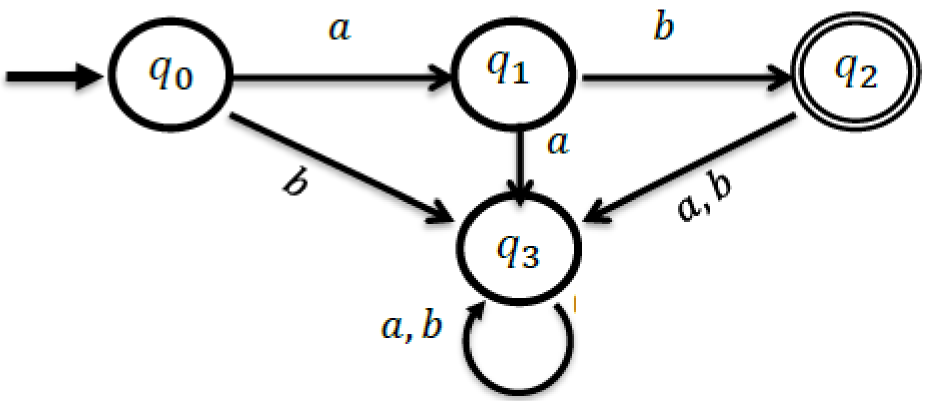
ما هو الأوتومات المنتهي الحتمي المقابل للتعبير المنتظم a^* علماً أن الأبجدية $\sum=\{a\}$



ما هو الأوتومات المنتهي الحتمي المقابل للتعبير المنتظم a^* علماً أن الأبجدية $\sum=\{a,b\}$

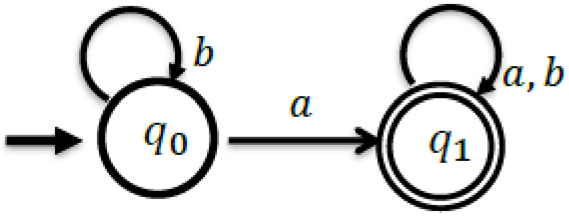


ما هو الأوتومات المنتهي الحتمي المقابل للتعبير المنتظم ab علماً أن الأبجدية ∑={a,b}

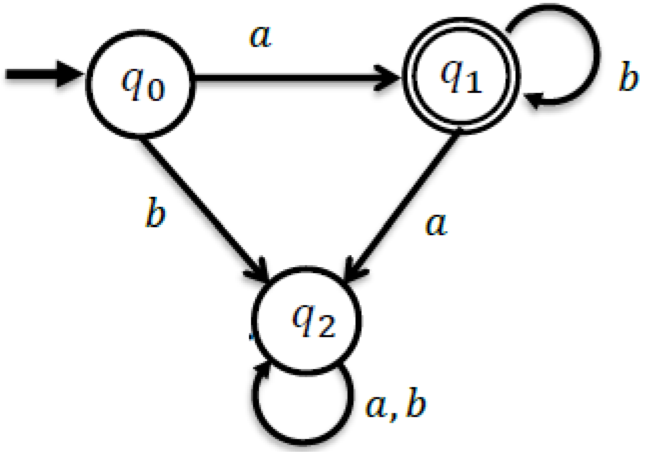


ما هو الأوتومات المنتهي الحتمي المقابل للتعبير المنتظم *(a+b) علماً أن الأبجدية ∑={a,b}

ما هو الأوتومات المنتهي الحتمي المقابل للتعبير المنتظم $b^*a(a+b)^*$ علماً أن الأبجدية $\sum=\{a,b\}$



ما هو الأوتومات المنتهي الحتمي المقابل للتعبير المنتظم ab^* علماً أن الأبجدية $\sum=\{a,b\}$



انتهت محاضرات الأسبوع 1