

٢٦ نيسان، ٠٨

مقدمة في PROLOG

أرائد الثالبي
ralthabti@aabfs.org

1

٢٦ نيسان، ٠٨

مقدمة في برولوج

- × PROLOG هي لغة برمجة للحاسب تعتمد مبادئ المنطق وهذا ما يفسر اسمها من الجيد ان نتعلم PROLOG لأن لها استخدام واسع في الابحاث وخاص الذكاء الصناعي وفي النظم الخبيره وقد تزايد استخدامها في التطبيقات التجارية.
- × تتكون PROLOG من قاعدة معطيات استاتيكية نصح عنها بدلا من لائحة التعليمات التي يعمل عليها الحاسب يتم حل المسائل بطرح اسئلة مصاغة بجذر وتبحث PROLOG عن الاجوبة في قاعدة المعطيات باستخدام الية بحث مدمجة ضمنها وهكذا تدعى PROLOG لغة تصريحية على عكس معظم اللغات الاخرى التي اجرائية
- × عندما نتعلم PROLOG سجد انها مختلفة تماما عن غيرها من اللغات الاجرائية واذا كنت قد برمجت من قبل فعليك ان تتعلم مجددا من البداية.
- × هناك امر يجب تذكره وهو انه بعكس الاعتقاد السائد فان PROLOG ليست غامضة او مبهمه بل تتبع مجموعة من القواعد كأي لغة اخرى.

2

٢٦ نيسان، ٠٨

أساسيات

- × سيرغب الكثير من الطلاب في البدء مباشرة بالبرمجة على الحاسوب لأن لغة PROLOG هي لغة برمجة في هذه المحاضرة سنتعلم كيف نستخدم لغة PROLOG لجعل الحاسب يقوم بأشياء نريدها.
- × لغة PROLOG محتواة ضمن برنامج يدعي بالمفسر INTERPRETER؛ لنرى كيف يمكنك تشغيل هذا المفسر لدخول والخروج من اللغة عندما ترغب .

3

٢٦ نيسان، ٠٨

الإقلاع

- × يمكنك الدخول إلى اللغة بكتابة كلمة prolog كما يحصل في معظم الأنظمة ومن بينها prolog LPA او Amzi IDE ثم ضغط مفتاح الإدخال . إذا كنت تستخدم PROLOG العامة انظر المقطع التالي . ستظهر لك علامة استفهام على الشاشة وبجوارها محث من الشكل:
- × -?
- × يمكنك الآن الكتابة وإدخال ما ترغب به لفعل ما تريد .
- × قد ترغب بمحاولة الخروج من مفسر لغة PROLOG اطبع.
- × halt. <ENTER> .-
- × لاحظ أن كلمة halt تبدأ بحرف h صغير إذ أن للأحرف الكبيرة أهمية خاصة في هذه اللغة لنتفق حالياً على استخدام الأحرف الصغيرة .
- × لاحظ أيضاً وجود نقطة بعد كلمة halt فكل شي تدخله في هذه اللغة يجب أن تلحقه بنقطة وعليك أن تتذكر ذلك دائماً قد تجد ذلك مزعجاً في بادئ الأمر وتنساه أحياناً لكن ستعود عليه لاحقاً.
- × بعد ضغط مفتاح الإدخال(ENTER) ستخرج من مفسر لغة PROLOG وقد تعلمت كيفية الدخول

4

الاستخدامات المتعددة للغة PROLOG

٢٦ نيسان، ٠٨

- × يجب أن نذكر أن مفسر prolog يختلف حسب الإصدار الذي تستخدمه .
التعليمات التي ذكرناها أعلاه خاصة بلغة LPA prolog وهي الأوسع في الملكية المتعددة، هناك إصدار أولي أساسي من هذه اللغة تم تطويره في جامعة Edinburgh ويُدعى أحياناً معيار Edinburgh ويشار إليه أحياناً باسم معيار clocksing mellish تحتوي معظم إصدارات prolog إضافات تتعدى هذا المعيار .
- × تعتبر LPA PROLOG ممتازة بالنسبة للإصدارات العامة وتستخدم في الجامعات والتطبيقات التجارية على حد سواء ويستخدم العديد من الطلاب لغة prolog PD لأنها مجانية وسهلة التعلم يجب أن نشير هنا أنه عليك كتابة Prolog PD لدخول المفسر في إصدار prolog PD وللخروج تكتب exitsys متبوعة بنقطة ثم مفتاح الإدخال

5

٢٦ نيسان، ٠٨

سؤال

- × ما هو الشيء الذي ينساه الجمع في بداية الامر!
- × الإجابة ينسى الجميع النقطة النقطة عند الإدخال.

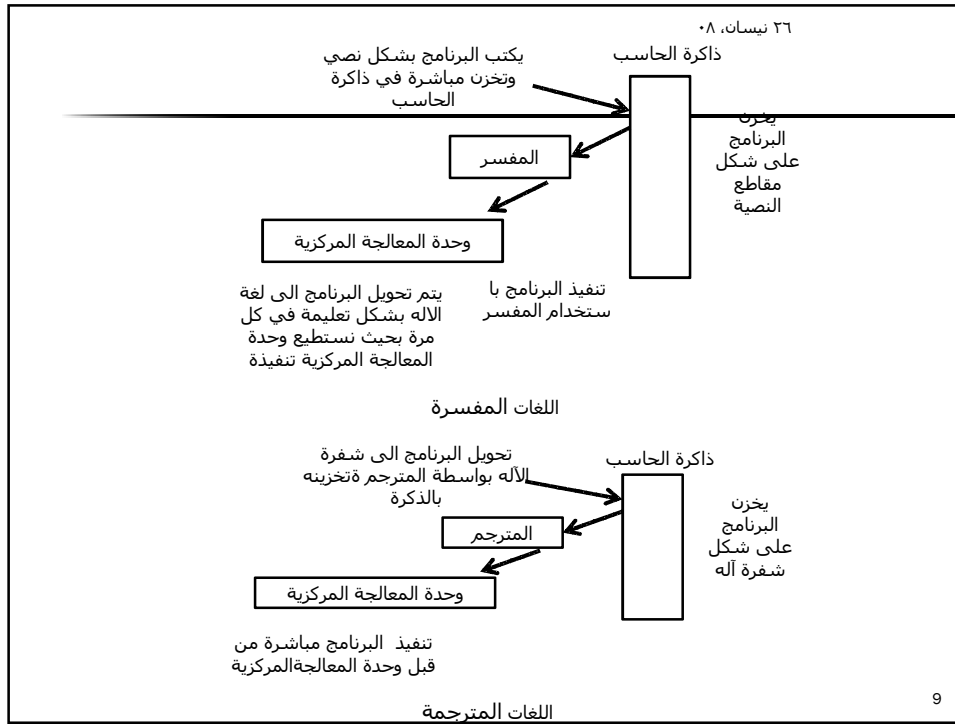
6

لغة PROLOG المفسرة

× ان prolog لغة مفسرة وهذا يعني ان كل شي تقوم بطباعته يتم ادخال ويفسر مباشرة الى صيغة يفهمها الحاسب ويعمل عليها من اللغات المفسرة الاخرى التي ربما استخدمتها لغة BASIC ولكن انتبه فهاتان اللغتان غير متشابهتين على الاطلاق هناك اختلاف جذري في الية عمل اللغات المفسرة واللغات المترجمة يحتاج الحاسب الى تحويل كل شيء الى ارقام ثنائية تدعي شفرة الآلة ليستطيع التعامل معها تتم طباعة البرنامج في اللغات المترجمة بشكل نص أو شفرة مصدرية ثم يجب ترجمته بالكامل الى لغة الآلة ليصبح برنامجا تنفيذيا و عندها يستطيع الحاسب تنفيذه في اللغات المفسرة لا يتم تحويل كامل البرنامج إلى شفرة الآلة القابلة للتنفيذ بل يحفظ بشكل شفرة المصدر كما هو ثم يتم تحويله إلى شفرة الآلة أثناء التنفيذ تعليمة تلو الأخرى . يقوم بعملية التحويل برنامج يدعى المفسر interpreter

سؤال

× ما هي برأيك محاسن ومساوئ كل من اللغات المترجمة والمفسرة؟
× تتطلب اللغات المترجمة وقتا طويلا لتطويرها لأن عليك الاستمرار بالترجمة لتتري ما إذا كانت تعمل ، لكنها اسرع اثناء التنفيذ وخالية من الأخطاء في معظم الأحيان لأن الأخطاء تظهر اثناء الترجمة ، اما اللغات المفسرة فهي اسهل واسرع اثناء التطوير لأنك تستطيع اختبار مقاطع قصيرة بدل عملية التحويل إلى شفرة الآلة تتم اثناء تنفيذ البرنامج ، وتعطي اخطاء اثناء ذلك . كذلك تحتل شفرة المصدر في اللغات المفسرة مساحة اكبر الحاسب من تلك التي تحتاجها الشفرة القابلة للتنفيذ في اللغات المترجمة



- × يظهر في الشكل لعلايه الفروقات بين اللغات المترجمة والمفسرة كما تلاحظ فان اللغات المترجمة تسبب الاريابك أثناء البرمجة أحيانا لكنها أفضل أثناء الاستخدام بينما تكون اللغات المفسرة أسهل أثناء البرمجة لكنها مربكة أثناء الاستخدام .
 - × لا تتكون لغة prolog من تعليمات لتشكيل البرنامج بل هي عبارة عن مجموعة عبارات هذه العبارات هي حقائق rules تشكل قاعدة معطيات .
 - × يقوم المفسر بتنفيذ عمليات على هذه الحقائق والقواعد استجابة لأهداف goals تعطيا له أنت .
 - × مازال بإمكانك كتابة برامج من النوع الذي يستخدم المصطلحات في prolog هذه البرامج تستطيع القيام بعمليات على من عدة أنواع سنتطرق خلال هذا الكتاب إلى الحقائق القواعد والبرامج .
 - × إذا كنت معتادا على المترجمة مثل Pascal و ADA فقد يلفت انتباهك أنك - في بعض إصدارات prolog (مثل LPA) - تستطيع ترجمة البرامج بعد كتابتها هذه العملية تحول البرامج إلى شفرة فعالة قابلة للتنفيذ خارج المفسر ولا يستطيع أي شخص آخر العبث بها .
 - × هذه العملية مفيدة في الإصدارات النهائية من البرامج التجارية ولكن أثناء تطوير البرامج فإن ميزة المفسر تكون سريعة ومفيدة .
 - × كما ستري لاحقا فإن هذه العملية التفسير تعني أنه بإمكانك فحص كل جزء من برنامج PROLOG بشكل منفصل .
- 10

القليل من الرياضيات

٢٦ نيسان، ٠٨

- * جرب طباعة <ENTER> 2=2-?
- * لا تنسى وضع النقطة وضغط مفتاح الإدخال .
- سنجد أن PROLOG يجب ب Yes لأن 2=2 حقا
- جرب <ENTER> apple=apple-?
- * لا تنسى وضع وضغط مفتاح الإدخال .
- * سيجيب prolog مجددا ب Yes لأنه من الواضح أن كلمة apple الأولى مشابهة تماما لكلمة apple الثانية . apple هنا هي ما يسمى في لغة prolog بالذرة apple أي مجموعة من الرموز الأبجدية . فالذرة في prolog هي بمثابة السلسلة في لغات البرمجة الأخرى.
- * ما يحدث في prolog هو أن المترجم يفحص كل ما يتم إدخاله ليرى إذا كان صحيحا (true) .
- * فإن كان كذلك يعيد الجواب ب Yes . جرب الآن .
- * <ENTER> 2+2 =4-?
- * لا تنسى وضع النقطة وضغط مفتاح الإدخال .
- * هذه المرة يعيد prolog الجواب ب no !! لأن إشارة = في prolog تستخدم لفحص التشابه من الناحية النظرية . ولأن 4 عدد و 2+2 تعبير رياضي سيفشل الاختبار . حاول إدخال ما يلي .
- * <ENTER> 2+2 is 4-?
- * لا تنسى وضع النقطة وضغط مفتاح الإدخال .
- * هذه المرة سيكون الجواب ب Yes لأن is تستخدم لحساب قيمة التعبيرات . و 2+2 تساوي 4
- * من الآن فصاعدا سنتوقف عن ذكر النقطة ولن نذكر بضغط مفتاح الإدخال على افتراض أنك تعلم بوجود القيام بهاتين العمليتين مع كل القيم التي تدخلها إلى محث prolog .

11

٢٦ نيسان، ٠٨

السؤال

- * ما هي إجابة prolog على كل مما يلي .
- * 100=100-? .
- * 1000/10=100-? .
- * 100 is 100/10-? .
- * 1000 is 100* 10-? .
- * 2 is (5+7)/6-? .
- * 74 is (5+7)*6-? .

12

٢٦ نيسان، ٠٨

الجواب

× $100 = 100 - 100 ?$

× Yes

× $1000 / 10 = 100 - 100 ?$

× no

× $1000 / 10$ is $100 - 100 ?$

× yes

× 1000 is 100 $10 - 100 ?$

× yes

× $6 / (5 + 7) = 2 - 2$ is

× $6 * (5 + 7)$ is $74 - 74 ?$

× yes

13

٢٦ نيسان، ٠٨

كتابة أشياء على الشاشة

× جرب ما يلي :-

× Write(jane).

× ستلاحظ طباعة الاسم Jane على الشاشة كما ستلاحظ طباعة Yes مباشرة بعد هذا الاسم إذا ما تراه هو .write(jane) ?

× Janeyes

× وهذا يبدو غريباً نوعاً ما . ما حدث هو أن write ليس عدداً ولا تعبيراً لذا يبحث عنه prolog في قاعدة معطيات . إن write هي كلمة خاصة في prolog تدعى مهلمات predicate وهي موجودة دائماً في قاعدة المعطيات وتقوم بكتابة وسائطها (argument) الجوده ضمن قوسين على الشاشة .

× إذا وجد العنصر ضمن قاعدة المعطيات وينجح في القيام بهمته يعتبر الـ write ناجحاً في prolog في هذه الحالة تتم طباعة الوسيط Jane - في هذه الحالة - على الشاشة

× ولأن write نجحت في القيام بهمتهما سيطلع prolog كلمة yes .

× ولكن شكل الطباعة غير مرتب بوجود yes مباشرة بعد الـ write جرب ما يلي

× $write(jane),nl$.

× هذه المرة ستبدو الاستجابة بالشكل

× $write(jane),nl$.

× Jane yes

× وهذه الشكل مرتب أكثر حيث أن nl هو معنن قياسي آخر في prolog ومعناه سطر جديد new line وهو موجود دوماً في قاعدة المعطيات وهدفة نقل السطر التالي .

× هناك نقطتان فائقتان الأهمية يجب ملاحظتهما هنا الأولى أننا باسـتخدام الفاصلة وضعنا حلين للاختبار في نفس العملية على أنها AND . والنقطة الهامة الثانية هي أن نتيجة كلا الاختبارين يجب أن تكون true حتى يطبع prolog - Yes وهذا محقق في حالة write وnl في مثالنا السابق لذا يأتي الجواب yes .

14

×	جرب ما يلي بأدخال خطأ كتابي :
×	?-write(jane),nx
×	لن يجد prolog (nx) في قاعدة المعطيات وسيفشل الاختبار ويعيد الجواب no أو predicate
×	(nx)not defined على الرغم من نجاح المعلن write وطباعة jane على الشاشة .
×	هناك طريقة أخرى لاستخدام write إذا كان الوسيط موجودا ضمن علاقي اقتباس فسوف تتم طباعة أي شيء بينهما . هذا يعني أنه يمكنك كتابة سلاسل تتضمن فراغات وأحرف كبيرة وليس ذرات فقط . جرب ما يلي :
×	?-write('Jane smith'),nl.
×	لو لم تكن Jane smith لعلاقي اقتباس لاعتبرت ذرة غير مقبولة بسبب وجود الفراغ والحرف الكبير في بدايتها وسيعطي prolog رسالة خطأ
×	السؤال
×	ماذا ستكون إجابة prolog ما يلي ؟
×	?-write(hello).
×	?-write(Hello).
×	?-write('Hello!').
×	?-write('Hello!'),nl.
×	
15	

٢٦ نيسان، ٨ إدخال عناصر إلى قاعدة المعطيات	
×	لنحاول وضع أسماء بعض الفتيات في قاعدة المعطيات . جرب أولاً :
×	?-girl(jane).
×	predicate (girl) not found أو no سيكون الجواب
×	وهذا يعني أنه تم البحث عن العنصر girl(jane) في قاعدة المعطيات ولم يتم إيجاده لنقم بوضعه هناك .
×	اكتب ما يلي بشكل متتال.
×	?-assert(girl(jane)).
×	?-assert(girl(sarah)).
×	?-assert(girl(norma)).
×	ذا كنت تستخدم prolog فعليك استخدام assertz بدلا من assert
×	معتبر المعلن assertz (أو assertz) هو أحد العنصرات القياسية في قاعدة المعطيات ويقوم بإضافة وسائمه . في هذه الحالة girl (jane) - باعتبار أنها عناصر جديدة في قاعدة المعطيات .
×	لا حظ طريقة طباعة زوجي الأقواس وألا فانك ستعرض لتبنيه بوجود خطأ قواعد المعطيات .
×	اطبع مرة ثانية.
×	?-girl(iane).
×	ستجد الآن أن الجواب هو Yes وهذا يدل على أنه تم إيجاد girl(jane) في قاعدة المعطيات . وقد نجح البحث . جرب أيضا girl(sarah) و girl(norma) وسنجد أن البحث سينجح وإذا أدخلت اسم فتاة غير موجود في قاعدة المعطيات مثل girl(heidi) فسوف يفشل البحث ويجب no لأن لعن girl معرف الآن .
×	يكنك رؤية ما هو موجود حاليا في قاعدة المعطيات بأدخال العبارة:
×	?-listing.
16	

- × هو معلن قياس آخر في prolog يطبع قاعدة المعطيات كاملة على الشاشة :
- × يقوم هذا المعلن بطباعة المعلنات القياسية مثل , write assert , nl لأنها موجودة دائما بل يكتب بطباعة ما قمت بأدخاله أنت .
- × يمكنك تخصيص العناصر لي تريد أن يقوم Listing بطباعتها ، عن طريق كتابة listing(girl)
- × هذا سيعطي نفس النتيجة في الوقت الحاضر لأنه لدينا عناصر girl فقط في قاعدة المعطيات . إذا كان هناك أنواع أخرى من العناصر في قاعدة المعطيات فسيقوم listing بالتكفل بأمر طباعتها أيضا .
- × عندما تخرج من prolog فإن كل العناصر المدخلة بهذه الطريقة ستزول . إذا أردت حذف أحد هذه العناصر مثل (sarah اكتب : retract.(girl(sarah))
- × سيقوم retract بإزالة girl(sarah) من قاعدة المعطيات وإذا طلب منه القيام بحذف عنصر غير موجود في قاعدة المعطيات فسوف يفضل .

17

٢٦ نيسان، ٠٨

الخلاصة

- × تتألف prolog من مفسر يأخذ المدخلات على المحث ويطبق عملياته على قاعدة المعطيات .
- × يمكن أن تعتبر المدخلات على المحث إما التعابير يختبر prolog صحتها أو خطأها ، أو أهدافا goals يبحث prolog عنها ليحدها أو لا وقد تكون مهامها تعطى لـ prolog لتنفيذها فينجح أو يخفق . دوما يعيد prolog yes أو no معطيا المجال لتقديم المزيد من الحلول إن كان ذلك مناسباً .
- × إن الوصول إلى الإجابة عن استفسار ما يتطلب البحث ضمن قاعدة المعطيات . إذا وجد prolog العنصر أو العناصر المعطية له ، وإذا نجح العنصر في تنفيذ مهمته المكلف بها فسيكون الجواب yes بعض المعلنات القياسية موجودة دائما في قاعدة المعطيات ويمكنك أيضا إضافة معلنات خاصة بك .

18

٢٦ نيسان، ٠٨

التكليف رقم (1)

- ×
× **تلكيف رقم ١**
× ١- جرب الأمثلة والتمارين الواردة في المحاضرة .
× ٢- اكتب مجموعة من المدخلات في محث prolog لوضع الحيوانات التالية في قاعدة المعطيات،
lion tiger cow
× ٣- ادخل استفسارا لمعرفة ما إذا كان هناك نمر (tiger) في قاعدة المعطيات .
× ٤- ادخل استفسارا لتعرف في نفس اللحظة ما إذا كان هناك lion و cow في قاعدة المطيات .
× ٥- قم الآن بإضافة مدخلات لتصنيف اثنين من الحيوانات على أنها لواحم الأسد والنمر .
× ٦- أنشئ واختبر استفسارا لمعرفة فيما إذا كان الأسد حيوانا ولاحما أيضا وما إذا كانت البقرة
حيوانا ولاحما أيضا .