

جلسه چهارم 93-8-27

برنامه‌سازی منطقی (10)

کتاب اول که از روی آن تدریس خواهد شد: Foundation of logic programming

دو فصل از این کتاب تدریس خواهد شد.

و بعد از این کتاب prolog programming For AI → Ivan Bratko

این کتاب ترجمه شده است

کاربردهای برنامه‌سازی منطقی

اثبات قضایا

زبانهای فرمال

زبانهای تابعی

پایگاه داده استنتاجی

پایگاه داده خودساز

پروگرامهای فازی

از زبانهای موجود در برنامه‌سازی منطقی Lisp است

که از سال ۱۹۷۰ به بعد مطرح شد

این پروگرام است که معروف تر و ساده تر است.

بافتات حجم منطقی است

Ch2

ادامه از اطلاعات صفحه 3 (دری دوم اطلاعات)

$P \rightarrow Q \equiv \neg P \vee Q$ (در پروگرام این طریقی)

$A_1 \vee A_2 \vee \dots \vee A_n \rightarrow B_1 \vee B_2 \vee \dots \vee B_k$

↓
* اگر تعداد متغیرها را کم کنیم هر دو عبارات اول درست می‌شوند.

$A_1 \vee A_2 \vee \dots \vee A_n \rightarrow (B_1 \wedge B_2 \wedge \dots \wedge B_k)$

↓ طبق دو جدول

و نتیجه می‌دهد

$A_1, A_2, \dots, A_n \leftarrow B_1, B_2, \dots, B_k$ * از راست به چپ

برای سادگی کار، and, or, و غیره لازم است. اما در سمت راست به معنی and

اما در سمت چپ به معنی or است
 $A_1, A_2, \dots, A_n \leftarrow B_1, B_2, \dots, B_k$

در اینجا سعی می‌کنیم بررسی کنیم (طبق این الیاد ~~در این الیاد~~ صفحه 3)

در اینجا می‌توانیم متلاشی کنیم؟

buff(bob) b_1
 pig(wilber) B_2

$buff(x) \wedge pig(y) \rightarrow \text{faster than } (x, y)$ B_3

Faster than (bob, wilber)
 A

چه فوایدی برایش زیر بریم.

$(b_1, b_2, b_3) \rightarrow A \leftarrow B_1, B_2, B_3$

A * از راست به چپ

Definite program (نیز معنی)

برای همین فوایدی که می‌بینیم در دو سمت معنی.

①

صنم مصاروم ۲۳۰۸۰۲۷

برنامه سازي منطق

محمد ابراهيم شيرى

دانشگاه صنعتى اميركبير - دانشكده رياضى و علوم كامپيوتر

بهار ۹۱-۹۲

کتاب مرجع :

① Foundations of Logic Programming

John Wylie Lloyd

② Prolog programming for artificial
intelligent

2

کتاب مفید دیگر :

- The art of prolog
sterling and shepiro
- Logic programming and prolog(2nd)
nilsson and maluszynski
- Programming in prolog
w.Clocksin and C.Mellish

ARTIFICIAL
INTELLIGENCE

② فصل از کتاب راه خوالیم

J.W. Lloyd

Foundations of
**Logic
Programming**

Second, Extended Edition



Springer-Verlag

3

تأليف
برجيه محمد داراد

PROLOG PROGRAMMING FOR ARTIFICIAL INTELLIGENCE

Ivan Bratko

E. Kardelj University · J. Stefan Institute
Yugoslavia

مركز البحوث والدراسات
العلمية
بجامعة
لبنان

William F. Clocksin
Christopher S. Mellish

Programming in Prolog

Using the ISO Standard
Fifth Edition



Springer

دانشگاه صنعتی امیرکبیر - دانشکده مهندسی کامپیوتر و علوم رایانه

4

* کاربرد های برنامه سازی منطقی

- اثبات قضایا
- زبان های فرمال
- زبان های تابعی (functional programming)
- پایگاه داده استنتاجی (deductive D.B.)
- پایگاه داده هوشمند
- برولوگ فازی

تاریخچه برنامه سازی منطق

- ❖ برنامه سازی منطق در سال 1970 شروع شد
- ❖ می توان گفت نتیجه کارهایی در رابطه با اثبات اتوماتیک قضایا در هوش مصنوعی است
- ❖ ایده اصلی بر می گردد به کارهای آقای kowalski در سال 1948
- ❖ در سال 1970 دانشمندان هوش مصنوعی به این نتیجه اساس دست یافتند که منطق را می توان در زبانهای برنامه سازی بکار گرفت
- ❖ در نتیجه زبانهای برنامه سازی منطق ایجاد گردید از جمله زبان Prolog
- ❖ Prolog در واقع ترکیب دو عبارت زیر است

Programming in LOGic

۱۵

فصل اول



تئوری مرتبه اول

تئوری مرتبه اول شامل موارد زیر است :

الفبا (alphabet)

زبان مرتبه اول (first order language)

مجموعه ای از قواعد بدیهی یا کلی (axiom)

مجموعه ای از قواعد استنباط یا نتیجه گیری (inference rule)

دهن سطح بر سر اول
در هر صورت طی کرد است

تئورک مرنبه اول

الفبا

الفبا شامل موارد زیر است : (هم الفبا شامل موارد زیر است)

- 1 متغیرها (variables) (u, v, w, x, y, \dots)
- 2 ثابت ها (constants) (a, b, c, \dots)
- 3 توابع (function symbols) (f, g, h, \dots)
- 4 علائم مسند یا خیری (predicate symbols) (P, Q, \dots)
- 5 ربط دهنده ها (connectives)
- 6 کمی کننده ها (quantifiers)
- 7 علائم نقطه گذاری و انشاء (punctuation symbols)

* نکته: موارد 1 تا 4 برای الفبا های مختلف ، متفاوت است
در حالی که موارد 5 تا 7 از الفبایی تا الفبای دیگر متفاوت است و موارد 2 و 3 ممکن است تهی باشند .

↓
در صورت نیاز باید

9

تئوری مرتبه اول

زبان مرتبه اول-قواعد بدیهی

* زبان مرتبه اول (first order language): زبان مرتبه

اول از جملات خوش تعریف

تشکیل شده (Well-Formed Formula) (WFF)

است.

قواعد بدیهی (axiom): قواعد بدیهی زیرمجموعه ای

از جملات خوش تعریف است.

نکته: از قواعد بدیهی و استنتاج برای به دست آوردن جملات جدید از روی جملات موجود استفاده می شود.

سید علی حسینی

الفبا

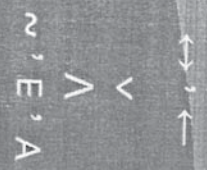
رابط دهنده-کمی کننده-علائم نقطه گذاری و انشاء

① ربط دهنده ها در منطق مرتبه اول :
 $\rightarrow, \leftrightarrow, \vee, \sim$

② کمی کننده ها در منطق مرتبه اول :
سور عمومی (A) ، سور وجودی (E)

③ علائم نقطه گذاری و انشاء :
" " ، " " ، " " ، " "

④ اولویت عملگرها :



75



علائمی برای نشان دادن خواص اشياء (properties) یا رابطه بین اشياء (relations) است.

Predicate

① Relation ; object \times object \rightarrow {T, F}
Love(Mary, Tom) \rightarrow {T, F}

علاقه

② Property : P \times object \rightarrow {T, F}
Fat(Tom) \rightarrow {T, F}

object
F یا T است یا

الفا

توابع

توابع Function : object* object --> object

مثال :

max's father → father (max)

max's father's wife → wife(father(max))

8

در هر جمله t_1, \dots, t_n در $Term$ زنجیره

کلمه (term) : هر کلمه به صورت بازگشتی به

صورت زیر تعریف میگردد :

هر متغیر یک ترم است

هر ثابت یک ترم است

اگر f یک تابع n تایی باشد و t_1, \dots, t_n ترم باشند آنگاه

if $t_1, \dots, t_n \Rightarrow f(t_1, \dots, t_n)$ is term

حمله (well-formed formula): هر جمله به صورت

بازگشتی به صورت زیر تعریف میگردد:

اگر p یک n -predicate باشد و t_1, \dots, t_n اترم باشد آنگاه $p(t_1, \dots, t_n)$

نیز یک جمله است (atomic formula).

اگر F و G دو جمله باشند آن گاه $F \wedge G$, $F \vee G$, $F \rightarrow G$ و

$F \leftrightarrow G$ هم جمله هستند. (Syntax)

اگر F یک جمله باشد و x نیز یک متغیر، آنگاه $\exists x F$ و $\forall x F$ نیز جمله

اند. (حملات سوربی)

نکته: از این به بعد برای هماهنگی با پرولوگ به جای $F \rightarrow G$

می نویسیم: $G \leftarrow F$ این اولین توسعه است. (توسعه خوب)

۹

منطق مرتبه اول زیر مجموعه ای از تئوری مرتبه اول است .

منطق مرتبه اول دارای دو جنبه است :

- نحو (syntax): به نحوه ساخت جملات خوش تعریف ،
توجه دارد
- مفهومی (semantic): به معنای جملات خوش تعریف ،
توجه دارد .

محدوده یا میدان دید (scope) در یک متغیر کمی

کننده ($\forall x$ و $\exists x$): محدوده متغیر، جمله ای است که

بلافاصله پس از آن می آید. یعنی در $\forall x F$ و $\exists x F$ ، F محدوده یا

میدان دید است

مثال: در جمله $\exists x p(x,y) \wedge q(x)$ محدوده $\exists x$ تنها $p(x,y)$

است در صورتی که در جمله $\exists x (p(x,y) \wedge q(x))$ محدوده $\exists x$

کل پرانتز $(p(x,y) \wedge q(x))$ است.

متغیر آزاد: متغیری که محدود نشده باشد یا کمی کننده ای روی

آن اعمال نشده باشد. مانند متغیر y در مثال های بالا