

Тема: Синтаксичний розбір виразів

Однієї з головних причин, що лежать в основі появи мов програмування високого рівня, з'явилися обчислювальні задачі, що вимагають великих об'ємів рутинних обчислень. Тому до мов програмування пред'являлися вимоги максимального наближення форми запису обчислень до природної мови математики. В зв'язку з цим однією з перших областей системного програмування сформувався дослідження способів виразів. Тут отримані численні результати, проте найбільше розповсюдження отримав метод трансляції за допомогою зворотного польського запису, який запропонував польський математик

Я

. Лукашевіч.

Нагадаємо, що існує цілий ряд методів синтаксичного розбору і обчислення виразів. Нехай вирази є рекурсивними

структурними

даними, які визначаються в термінах самих себе.

Якщо ви обмежитеся використанням у виразах тільки операторів +,

-

* / і дужок

,

то зможете сказати, що всі вирази можуть бути визначеними в термінах наступних правил:

выраз=>терм[+терм][-терм]

терм=>множник[*множник][/множник]

множник=>змінна, число або (вираз)
де будь-яка частина може бути порожньою.

Квадратні дужки

позначають

необов'язковість,

$a \Rightarrow$ - утворення. Фактично правила є правилами утворення виразів.

Вираз

$10+5*B$

складається з двох термів: 10 і $5*B$. Проте, включає три множники:

10,

5 і B.

Цими множниками є два числа і одна змінна.

З другого боку вираз

$14*(7-C)$

має два

терми 10 і $(7-C)$,

один з яких є числом,

а
інша дочірнім виразом. Дочірній вираз розпадається на одне число і одну змінну.

Даний процес формує основу для рекурсивного низхідного синтаксичного

розбору,

який є набором загальних

рекурсивних процедур,

що носять характер ланцюжка. На кожному відповідному кроці

синтаксичний розбір може виконувати задані операції в алгебра правильній послідовності. Наприклад розглянемо синтаксичний розбір вхідного виразу $9/3-(100+56)$ і виконання операцій по кроках.

Крок 1. Узяти першу лексему: 9/3

Крок 2. Узяти обидва множники і виконати операцію поділу.

В результаті виходить 3.

Крок 3. Узяти другу лексему: (100+56). В даній точці ви повинні рекурсивно проаналізувати другий вираз.

Крок 4. Узяти обидва числа і скласти. В результаті виходить 156.

Крок 5. Повернутися з рекурсивного виклику і відняти 156 з 3, що дає відповідь - 153.

Якщо ви трохи заплуталися, не турбуйтеся. Це складна концепція. Потрібно засвоїти

два моменти про даний рекурсивний погляд на

вирази: по-перше, передування операторів є неявним при заданих правилах породження виразів; по-друге, даний метод синтаксичного розбору і обчислення виразів дуже схожий на

те, як це робиться уручну.

Рядкові величини

- операція + (з'єднання) 'місто'+ ' '+ 'Луцьк'

Функції роботи з рядками:

№

Назва функції

Призначення

Приклад

Результат

Синтаксичний розбір виразів

Добавил(a) Administrator
22.04.11 14:09 -

1.

Length(S)

визначає кількість символів у заданому рядку

Length ('місто Луцьк')

11

2.

Copy(S,n,m)

виділяє m символів рядка S, починаючи від символу з номером n

Copy ('місто Луцьк', 6, 5)

'Луцьк'

3.

Синтаксичний розбір виразів

Добавил(a) Administrator
22.04.11 14:09 -

Pos(S1, S2)

визначає номер символу, з якого починається входження рядка (тексту) S1 у рядок S2

Pos (' ', 'місто Луцьк')

6

4.

Concat(S1, S2,...)

з'єднує рядки в один рядок

Concat('20', '01')

'2001'

Процедури роботи з рядками:

№

Назва функції

Призначення

Приклад

Результат

1.

Insert (A:string, var B: string, n:integer)

вставляє рядок A у рядок B, починаючи від позиції з номером n

S1:='місто';

S2:='Луцьк';

Insert(S1,S2,1);

Синтаксичний розбір виразів

Добавил(a) Administrator
22.04.11 14:09 -

☐ 'містоЛуцьк';

☐

☐ 2.

☐ Delete (var S:string, n:integer, m:integer)

☐ вилучає m символів з рядка S, починаючи від позиції n

☐ S:= 'містоЛуцьк';

☐ delete(S,1,5);

☐ 'Луцьк';

☐

☐ 3.

☐ Str (A:integer, var S:string)

Синтаксичний розбір виразів

Добавил(a) Administrator
22.04.11 14:09 -

переводить числове дане A у дане типу рядок

A:=2001;

Str(A,S);

'2001'

4.

Val (S: string, var A, KOD: integer)

засилає у числову змінну A числовий образ рядка S, повертаючи код помилки KOD

S:='2001';

Val(S,A,Kod);

2001