



Tutoriaal 9 Gebruik van LEX en YACC

Rekenaarwetenskap 324 Teoretiese Rekenaarwetenskap

12 Mei 2004

1 Doel van die tutoriaal

Na hierdie tutoriaal behoort u in staat te wees om:

- 'n Klein sintaksontleder met behulp van YACC te genereer.
- 'n Leksikale ontleder met behulp van LEX te skep wat dan deur u sintaksontleder gebruik kan word.

2 Beskrywing / Opdrag

Skryf 'n LEX/YACC program wat 'n eenvoudige sakrekenaar ("desktop calculator") implementeer. Dié rekenaar moet aan die volgende vereistes voldoen:

- Dit beskik oor 26 registers genaamd a tot z . Die gebruiker kan 'n waarde aan 'n registers toeken. Byvoorbeeld

```
>> a = 2
```

ken die waarde 2 aan die register toe, terwyl

```
>> a
```

die waarde van a op die skerm gaan vertoon wat in hierdie geval 2 is. Die waarde van die register kan natuurlik die resultaat van 'n komplekse berekening wees.

- Die rekenaar moet die volgende wiskundige operasie implementeer:

$$\{+, -, \times, \div, \wedge, \sin, \cos, \tan, !\}.$$

Let op dat \wedge die magsverheffingsoperator is en $!$ die fakulteitsoperator. Verder moet die sakrekenaar onderskei tussen die voorrange van die verskillende operators. Let op dat \wedge die hoogste voorrang het en dat dit regsassosiatief is. Ook moet die rekenaar tussen unêre minusse en die gewone binêre minus kan onderskei. Daar moet ook voorsiening gemaak word vir hakies in uitdrukkings en die nodige sorg moet aan die dag gelê word ten einde te verseker dat hakies die hoogste voorrang geniet.

- Indien 'n vergelyking te lank is, kan die gebruiker ... intik en op die volgende lyn voortgaan met die vergelyking. (Wenk: Dit word baie maklik in die leksikale ontleder gedoen.)
- Die rekenaar moet kommentaar wat met 'n % begin, ignoreer. Die kommentaar eindig aan die einde van die lyn waarop dit verskyn.
- Waardes wat ingetik word kan gewone integers, sowel as wisselpuntgetalle wees. Alle interne berekeninge moet in dubbelpresisie wisselpuntformaat ("double" in C), gedoen word.
- Die sakrekenaar moet kassensitief wees en moet witspasie in uitdrukkings ignoreer.
- Ook moet die volgende bevele herken word:
 - *quit* - Dit moet die sakrekenaarprogram termineer.
 - *clear all* - Stel die waarde van al die interne registers gelyk aan nul.
 - *clear < register >* - Stel die inhoud van register *< register >* gelyk aan nul.
- Die program moet nie termineer indien die gebruiker 'n foutiewe uitdrukking invoer nie. Dis nie nodig om ingewikkelde fouthanteringsmeganismes te implementeer nie – slegs 'n generiese boodskap dat die invoer verkeerd is en dan die geleentheid om dit oor in te tik, is voldoende.

Hier is 'n voorbeeld van 'n tipiese sessie met die rekenaar. >> verwys telkemale na die invoer van die gebruiker, terwyl die afvoer van die rekenaar op die volgende lyn gaan volg.

```
>> x = 2 % Stel die waarde van x = 2
>> y = 3 % Stel die waarde van y = 2
>> z = 3*y + x
>> z
11.0000
>> 100+200*10
2100.0000
```

```

>> sin(300*0)
0.0000
>> 3 + 5 + ...
> 2
10.0000
>> 3 + 2 +
Parse error
Please re-enter the last line
>> 3 + 2
5.0000
>> 1/5
>> 0.2000
>> clear all
>> x
0
>> y
0
>> z
0
>> quit

```

Belangrik

Die naam van u LEX spesifikasie moet *calc.l* wees en die naam van die YACC program *calc.y*.

3 Wenke en/of ekstra hulp

Op die web, is daar 'n *zip* lêer beskikbaar genaamd, *yacctut.zip*. Hierdie lêer bevat template om u te help om aan die gang te kom. Daar word ook 'n *makefile* verskaf. Sorg dat hierdie lêer in dieselfde gids is as al u ander bronkode lêers. Om u program te vertaal, tik

```
$ bison -d calc.y
```

en verander

```
typedef int YYSTYPE;
```

na

```
typedef double YYSTYPE;
```

in die header file *calc.tab.h* indien nodig.

Tik nou

```
$ make
```

Dit sal die nodige programme roep en 'n uitvoerbare program *calc* skep.

Inhandiging

Vrydag 21 Mei, 14:00. Plaas *calc.l* *calc.y* en *calc* in *rw324/tut9*.