

CoCoA-5 - Design #1540

Double power

16 Nov 2020 18:05 - John Abbott

Status:	Closed	Start date:	16 Nov 2020
Priority:	Normal	Due date:	
Assignee:	John Abbott	% Done:	100%
Category:	enhancing/improving	Estimated time:	2.49 hours
Target version:	CoCoA-5.4.0	Spent time:	2.35 hours
Description What do you think CoCoA does with the following input? <code>x^2^3;</code> What do you think CoCoA should do with that input?			
Related issues:			
Related to CoCoALib - Design #1538: RingElem from string (ReadExpr)		Closed	13 Nov 2020
Related to CoCoALib - Bug #1579: Readexpr/RingElem: unhelpful error message w...		Closed	23 Feb 2021

History

#1 - 16 Nov 2020 18:07 - John Abbott

I think CoCoA should give a warning or even an error. Currently, it gives neither!
On my computer I got `x^8`

Since CoCoA warns for something like `2/3*x`, I was quite surprised that no warning was given for a "double power".

#2 - 17 Nov 2020 13:19 - John Abbott

- Status changed from New to In Progress
- % Done changed from 0 to 10

If we do look at handling exponents, it might be nice to improve the error mesg when a negative exponent is given without brackets:

```
/**/ 2^-2;  
--> ERROR: Invalid start of expression  
--> 2^-2;  
--> ^
```

It would be more helpful if the message said that negative exponents must be in brackets (or indeed any exponent which is not a non-neg integer literal).

#3 - 20 Nov 2020 11:39 - John Abbott

- Related to Design #1538: RingElem from string (ReadExpr) added

#4 - 20 Nov 2020 14:11 - John Abbott

I would like to maintain reasonable compatibility between the way CoCoA-5 parses expressions and the way RingElem or ReadExpr parses expressions.

In particular, on the whole, if an expression is valid for CoCoA (without warnings/errors) then it should be valid with the same meaning in ReadExpr; and *vice versa*.

An significant difference at the moment is that CoCoA-5 allows fairly general expressions as exponents whereas ReadExpr expects just an integer literal (in brackets if negative).

#5 - 23 Dec 2020 11:37 - John Abbott

Relevant source is probably in Parser.C around line 1890; **Parser::parsePowerExpression**

#6 - 22 Jan 2021 10:31 - John Abbott

- Target version changed from CoCoA-5.4.0 to CoCoA-5.4.2

#7 - 17 Feb 2021 18:33 - John Abbott

- Status changed from In Progress to Resolved

- Assignee set to John Abbott

- % Done changed from 10 to 70

I think I have solved the main problem and also the one mentioned in comment 2.
Solution is not elegant, but I do not want to spend more time working with the parser.

Will check in soon.

Current behaviour:

```
/**/ 2^3^4;
--> WARNING: "^" is right associative; use brackets to avoid this warning
--> 2^3^4;
-->   ^^^
2417851639229258349412352
/**/ 2^+2;
--> ERROR: Exponent may not start with plus or minus; use brackets
--> 2^+2;
-->   ^
/**/ 2^-2;
--> ERROR: Exponent may not start with plus or minus; use brackets
--> 2^-2;
-->   ^
```

#8 - 23 Feb 2021 10:50 - John Abbott

- *Related to Bug #1579: Readexpr/RingElem: unhelpful error message when input is wrong added*

#9 - 24 Sep 2021 21:43 - John Abbott

- *Status changed from Resolved to Closed*

- *% Done changed from 70 to 100*

- *Estimated time set to 2.49 h*

#10 - 24 Sep 2021 21:45 - John Abbott

- *Target version changed from CoCoA-5.4.2 to CoCoA-5.4.0*