CoCoALib - Bug #1579

Readexpr/RingElem: unhelpful error message when input is wrong

23 Feb 2021 10:44 - John Abbott

Status:	Closed	**Start date:**	23 Feb 2021
Priority:	Low	**Due date:**	
Assignee:	John Abbott	**% Done:**	100%
Category:	Improving	**Estimated time:**	3.01 hours
Target version:	CoCoALib-0.99850	**Spent time:**	3.10 hours

Description

I find the following error message unhelpful

```
/**/ RingElem(QQ, "2^3^4");
--> ERROR: Extra chars after ringelem expr
--> [CoCoALib] ReadExpr(string)
--> RingElem(QQ, "2^3^4");
--> ^^^^^^^^^^^^^^^^^^^^^
```

This issue was partly inspired by issue #1540 and also #1538.

Related issues:

Related to CoCoALib - Design #1538: RingElem from string (ReadExpr)	**Closed**	**13 Nov 2020**
Related to CoCoA-5 - Design #1540: Double power	**Closed**	**16 Nov 2020**

History

#1 - 23 Feb 2021 10:49 - John Abbott

- Category set to Improving

- Priority changed from Normal to Low

- Target version set to CoCoALib-0.99800

Currently permitted expressions require that an exponent be an unsigned integer literal or a possible signed integer literal inside (round) brackets.
It is probably not worth trying to allow more general exponents.

Since CoCoA-5 interprets the exponentiation operator ("hat") as right-associative, "ReadExpr" should not treat it as left-associative.
So RingElem(QQ,"2^3^4") must give an error; but the actual message produced could be more helpful.

What should the error mesg say?

#2 - 23 Feb 2021 10:49 - John Abbott

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

#3 - 23 Feb 2021 10:50 - John Abbott

- Related to Design #1540: Double power added

#4 - 15 Apr 2021 10:38 - John Abbott

- Status changed from New to In Progress

- Assignee set to John Abbott

- % Done changed from 0 to 10

I have added a simple extra test: if the first char after the end of the expr is '^', then it prints out a more helpful message:

```
/**/ RingElem(QQ, "2^3^4");
--> ERROR: Double exponent not allowed; use brackets like (a^b)^c
--> [CoCoALib] ReadExpr(string)
--> /**/ RingElem(QQ, "2^3^4");
-->      ^^^^^^^^^^^^^^^^^^^^^
```

Are there any other special cases?
Perhaps the general message should at least say what the first unexpected char is?

How about **RingElem(P, "(x+y)(x-y)")**?
Also **RingElem(P,"2x")**?
Perhaps if the extra char is one which could appear after a + or *, the err mesg could suggest that an operator is missing.
Such chars are digits, alpha, open-bracket.
An excess close-bracket could suggest that an open-bracket was missing earlier...

Mmmm, not so easy to guess what the mistake might be.

#5 - 03 Nov 2021 16:58 - John Abbott

- Target version changed from CoCoALib-0.99800 to CoCoALib-0.99850

- % Done changed from 10 to 30

#6 - 21 Jan 2024 20:16 - John Abbott

- % Done changed from 30 to 80

I have added a new error mesg for the case that the first extra char is open-bracket or letter or digit.
The err mesg append a hopefully helpful comment "missing operator?"

```
/**/ RingElem(R,"2x");
--> ERROR: Extra chars (first is 'x') after ringelem expr -- missing operator?
--> [CoCoALib] ReadExpr(string)
--> RingElem(R,"2x");
--> ^^^^^^^^^^^^^^^^
/**/ RingElem(R,"(x+1)(x-1)");
--> ERROR: Extra chars (first is '(') after ringelem expr -- missing operator?
--> [CoCoALib] ReadExpr(string)
--> RingElem(R,"(x+1)(x-1)");
--> ^^^^^^^^^^^^^^^^^^^^^^^^
```

I'm not too convinced by the parenthetical comment (first is `(`). Any better ideas?

#7 - 25 Jan 2024 10:12 - John Abbott

Here is another test case: RingElem(R, "y(x)")
Looks like a function call...

#8 - 25 Jan 2024 16:21 - Anna Maria Bigatti

- Subject changed from Readexpr/RingElem: unhelpful err mesg when input contains double power to Readexpr/RingElem: unhelpful error message when input is wrong

I think I improved these error messages (hard to spot in string otherwise):

```
RingElem(CurrentRing, "18*x^4*y*z +720*x^3*y^2*z -135*x^4z^2 -10*x^3*y*z^2");
--> ERROR: Unexpected char 'z' in string from "..z^2 -10*x^.." -- maybe forgotten '*'?
--> [CoCoALib] ReadExpr(string)
--> RingElem(CurrentRing, "18*x^4*y*z +720*x^3*y^2*z -135*x^4z^2 -10*x^3*y*z^2");
--> ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

RingElem(CurrentRing, "18*x^4*y*z +720*x^3*y^2*z -135*xx^4z^2 -10*x^3*y*z^2");
--> ERROR: symbol "xx" not in ring
--> [CoCoALib] ReadFactor: symbol
--> RingElem(CurrentRing, "18*x^4*y*z +720*x^3*y^2*z -135*xx^4z^2 -10*x^3*y*z^2");
--> ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

What confused me at the start was that the "real" CoCoA interpreter is so much better is indicating these errors.

#9 - 25 Jan 2024 16:26 - Anna Maria Bigatti

Still to do: check other cases.
I modified only ReadExpr(string) and ReadAtom (for symbol).
I prefer to document this and check it in before I forget.

#10 - 23 Feb 2024 12:39 - Anna Maria Bigatti

- Status changed from In Progress to Closed

- % Done changed from 80 to 100

- Estimated time set to 3.01 h

Anna Maria Bigatti wrote:

> Still to do: check other cases.

I cannot remember what I meant.
Now seems pretty good, and in case we'll make a new issue.
Closing.