

rethrow in Try .. UponError .. EndTry

| | | | |
|------------------------|--------------------|------------------------|-------------|
| Status: | New | Start date: | 08 Mar 2012 |
| Priority: | Normal | Due date: | |
| Assignee: | | % Done: | 0% |
| Category: | Parser/Interpreter | Estimated time: | 0.00 hour |
| Target version: | CoCoA-5.?.? | Spent time: | 0.90 hour |

is it possible to have a better location or all the information is lost when using Try?

| | | |
|--|--------|-------------|
| Related to CoCoA-5 - Bug #170: CoCoAManual HTML: page for "Try" | Closed | 25 May 2012 |
| Related to CoCoA-5 - Bug #287: --fullCoCoALibError prints too much (Try/UponE... | New | 12 Dec 2012 |

1/3

#3 - 10 Dec 2012 15:30 - John Abbott

What is the correct way to write a CoCoA-5 test which checks that a certain call produces the expected error? It should complain if an error different from the expected one is encountered.

#4 - 12 Dec 2012 10:43 - Anna Maria Bigatti

What about this?

(note that, as said above, all other error information is lost :-()

```
Define CheckErrMesg(F, Arg1, SubstringErrMesg)
  Try IgnoreValue := F(Arg1);
  UponError e Do
    If Not (SubstringErrMesg IsIn GetErrMesg(e)) Then
      Error("EXPECTED Error: " + GetErrMesg(e));
    EndIf;
  EndTry;
EndDefine; -- CheckErrMesg

CheckErrMesg(deg, 0*x, "Non-zero");
CheckErrMesg(deg, 0*x, "aksdhfkl");
```

Note: this is valid only for **functions** (not procedures) which expect just 1 arg. Anna suggests modifying it to accept a list of args (up to length 3?), and then use a cascade of **If** to call the function in the right way.

2012-12-12 JAA nice idea -- let's try it!

#5 - 02 Apr 2014 17:34 - Anna Maria Bigatti

- Target version set to CoCoA-5.1.0 Easter14

#6 - 09 Apr 2014 17:27 - John Abbott

- Target version changed from CoCoA-5.1.0 Easter14 to CoCoA-5.?.?

#7 - 14 Oct 2020 13:19 - John Abbott

Have we used the idea in comment 4 in the CoCoA-5 tests?

I think probably not. If not, why not?

Solving the original problem looks like it would be hard and lengthy... I don't want to try :-(

I also wonder whether it is correct to solve the original problem. It might be slightly helpful during debugging, but otherwise...??

#8 - 23 Oct 2020 10:18 - John Abbott

- *Subject changed from Try .. UponError .. EndTry to rethrow in Try .. UponError .. EndTry*