CoCoA-5 - Feature #101

rethrow in Try .. UponError .. EndTry

08 Mar 2012 12:13 - Anna Maria Bigatti

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

Description

What is the way to deal with errors we don't want to catch? See this example:

```
Define MyDeg(F)
  Try
    D := Deg(F);
  Return D;
UponError E Do
    If "Non-zero RingElem required" IsIn GetErrMesg(E) Then
      Return -123456;
Else
    error(GetErrMesg(E));
EndIf;
EndTry;
EndDefine;
```

So MyDeg(zero(R)) is caught and MyDeg("a") is an error, but

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

Related issues:

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

History

#1 - 08 Mar 2012 15:49 - Giovanni Lagorio

I'm afraid that information is lost at the moment (the error object contains the error message only, more information should be embedded there to allow a later re-throw of the "same" error)

#2 - 09 Mar 2012 11:02 - Anna Maria Bigatti

Giovanni Lagorio wrote:

(the error object contains the error message only, more information should be embedded there to allow a later re-throw of the "same" error)

Can you estimate how difficult it would be to add the information? i.e. is there any hope for me to make it?

09 Apr 2024 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, Argl, SubstringErrMesg)
   Try IgnoreValue := F(Argl);
   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...??

09 Apr 2024 2/3

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

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

09 Apr 2024 3/3