

## CoCoA-5 - Bug #1095

### Timeout not working as hoped/expected

29 Aug 2017 16:29 - John Abbott

<b>Status:</b>	Closed	<b>Start date:</b>	29 Aug 2017
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	John Abbott	<b>% Done:</b>	100%
<b>Category:</b>	enhancing/improving	<b>Estimated time:</b>	1.11 hour
<b>Target version:</b>	CoCoA-5.2.2	<b>Spent time:</b>	1.10 hour
<b>Description</b>			
I had hoped to use the "time out" feature in a loop similar to this:			
<pre>for i := 1 to 1000 do   I := RandomIdeal(...);   try     GB := GBasisTimeout(I, 5.0);   UponError E do /*nothing*/   EndTry;   ... endfor;</pre>			
Unfortunately as currently implemented the "time out" causes an interrupt (like Ctrl-C) rather than a normal error; so control leaves the <b>try</b> block whenever a time out occurs...			
Discuss; fix.			
<b>Related issues:</b>			
Related to CoCoALib - Feature #385: Design new errors using inheritance		<b>In Progress</b>	<b>08 Jul 2013</b>

### History

#### #1 - 30 Aug 2017 17:11 - John Abbott

- Description updated

Is a "timeout" more like an interrupt (which cannot be caught by a **try** block)?  
Or is it more like an error (which can be caught by a **try** block)?

In my example I wanted it to behave like an error.  
Was my desire correct?

#### #2 - 30 Aug 2017 17:32 - Anna Maria Bigatti

- % Done changed from 0 to 10

John Abbott wrote:

Is a "timeout" more like an interrupt (which cannot be caught by a **try** block)?  
Or is it more like an error (which can be caught by a **try** block)?

My first thought was it should be like an interrupt ("interrupt after ..seconds"),  
but I see it could be viewed similarly to the "insufficient precision" error.

The question is: how can we detect the computation finished?

One easy way is this (probably easier than try/catch for the common user):

```
UnsetValue := 145692761358726487564386; -- whatever object
GB := UnsetValue;
GB := GBasisTimeout(I, 10);
if GB <> UnsetValue then ...
```

### #3 - 01 Sep 2017 11:28 - John Abbott

- Related to Feature #385: Design new errors using inheritance added

### #4 - 06 Sep 2017 15:17 - John Abbott

- Status changed from New to Resolved

- Assignee set to John Abbott

- % Done changed from 10 to 80

I have modified the code so that **TimeoutException** is now derived from **ErrorInfo** (rather than **InterruptException**). This seems to make everything work as expected (based on just 1 test).

I'll check in shortly.

### #5 - 13 Nov 2017 15:04 - John Abbott

- Status changed from Resolved to Closed

- % Done changed from 80 to 100

- Estimated time set to 1.11 h

I have not really tested this, but think that the current impl is OK anyway ==> closing.