

CoCoA-5 - Bug #1239

CartesianProductList: case of just 1 entry in list

29 Jan 2019 12:09 - John Abbott

Status:	Closed	Start date:	29 Jan 2019
Priority:	High	Due date:	
Assignee:	Anna Maria Bigatti	% Done:	100%
Category:	bug	Estimated time:	0.99 hour
Target version:	CoCoA-5.3.0	Spent time:	1.00 hour
Description			
I believe CartesianProductList handles wrongly the case of a list with just 1 entry:			
<pre>>>> CartesianProductList([0..1, 0..1, 0..1]); [[0, 0, 0], [0, 0, 1], [0, 1, 0], [0, 1, 1], [1, 0, 0], [1, 0, 1], [1, 1, 0], [1, 1, 1]] >>> CartesianProductList([0..1, 0..1]); [[0, 0], [0, 1], [1, 0], [1, 1]] >>> CartesianProductList([0..1]); [0, 1]</pre>			
Surely the last result ought to be [[0], [1]] , right?			

History

#1 - 29 Jan 2019 12:11 - John Abbott

The bug would found by a student in Passau (Bernhard Andrashko).
His program misbehaved when computing CartesianProductList with a single "factor".

#2 - 15 Feb 2019 12:21 - Anna Maria Bigatti

- Assignee set to Anna Maria Bigatti

- % Done changed from 0 to 10

hmmmm, true.

I had been reasoning too mathematically....

I do not much like it though.

should instead return an error? (or maybe just print a warning?)

Do you remember how that was used?

Anyway, I have fixed it in my copy, now.

#3 - 15 Feb 2019 12:50 - John Abbott

- Status changed from New to In Progress

- % Done changed from 10 to 20

I can try asking the student again for a copy of his program. However, the student was fairly good at programming, and I was mystified when he inserted code to handle a "special case" when I fully expected the general case to work (but it did not because of the bug in this issue).

Given my expectation, I think the case of `CartesianProductList(L)` where `L` is a list of a single list should work without warning (and certainly it should not give an error). While I understand the thinking behind "protecting a programmer from a careless mistake", I am not happy about a design which makes it more difficult to write a correct program (*i.e.* needing special handling for the case of a cartesian product of 1 factor). I also note that the function `product` works fine when given as argument a list containing a single entry.

#4 - 15 Feb 2019 12:54 - John Abbott

I do note that `CartesianProductList([])` returns `[]` without warning (or error, obviously).

I **might** be willing to be convinced that this case ought to produce a warning/error...

#5 - 15 Feb 2019 14:37 - Anna Maria Bigatti

- % Done changed from 20 to 50

OK, convinced.

If he actually wrote a program where he needed to handle the 1-case (and had to design his own workaround) then we fix it. Already done.

For the empty list I'm flexible. Now I think that the 0-case is OK and should give *no nasty surprises*:

```
/**/ CartesianProductList ([[ ], [ ], [ ]]);  
[]  
/**/ CartesianProductList ([ ]);  
[]
```

#6 - 15 Feb 2019 14:49 - John Abbott

- Status changed from *In Progress* to *Resolved*

- % Done changed from 50 to 80

Good, we're agreed. I've fixed it too -- we'll merge when I'm in Genoa.

I do still have a minor doubt about the empty product, but it's not that important.

#7 - 04 Mar 2019 12:30 - John Abbott

- Status changed from *Resolved* to *Closed*

- % Done changed from 80 to 100

- Estimated time set to 0.99 h

JAA notes that the product of zero factors should return a list containing 1 element which is a 0-tuple (namely, the list of the empty list). Code corrected.

Checked in. Closing.