

CoCoA-5 - Feature #95

Error/warning if fn-proc is neither fn nor proc?

29 Feb 2012 12:38 - John Abbott

Status:	New	Start date:	29 Feb 2012
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Parser/Interpreter	Estimated time:	0.00 hour
Target version:	CoCoA-5.?.?	Spent time:	0.25 hour
Description My email records show that we wrote about this on 2010-05-19. C5 distinguishes fns from procs simply by the nature of the return statements appearing the definition. An parse-time error is signalled if there are two return statements of differing type. At the moment no parse-time error (or warning) is given if a fn could "fall off the end" (but a run-time error is generated should execution actually fall off the end). We opted not to give any warning or error as it is difficult to do correct code analysis to determine when the source code is plausible, and the alternative is to force the user to insert annoying, needless return statements just to keep the parser quiet. I now favour issuing a warning (at least), because a student supplied me with some code which did indeed have a missing final return. If I recall well, we have several levels of warnings, so maybe this warning could be (initially) issued only at the highest level (to avoid annoying too many people).			
Related issues: Related to CoCoA-5 - Feature #492: New C5 fn for adjusting the warning level			
		New	24 Mar 2014

History

#1 - 31 May 2012 18:09 - Anna Maria Bigatti

- Category set to Parser/Interpreter

#2 - 02 Apr 2014 17:34 - Anna Maria Bigatti

- Target version set to CoCoA-5.1.0 Easter14

#3 - 09 Apr 2014 17:24 - John Abbott

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

#4 - 15 Jun 2016 14:13 - John Abbott

Here is a simple test case:

```
define f(x)
  if x > 0 then return 1; endif;
enddefine; -- f
```

Here is another test case:

```
define f(x)
  n := 0;
  while true do
    incr(ref n);
    if n > x then return n; endif;
  endwhile
enddefine
```