Anubis - Feature #19
Add default value for "force_nth()" and remove "alert()"

09/23/2007 01:09 PM - Anonymous

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Alain Prouté	**% Done:**	0%
Category:	Libraries	**Estimated time:**	0.00 hour
Target version:	1.8		
Platform:		**Triage Stage:**	
Resolution:	fixed		

Description

The function *force_nth* is very dangerous and doesn't correspond to the Anubis philosophy (in my mind).
So it would be better to provide a default value for $T when calling it, and remove the *alert* call.

History

#1 - 09/23/2007 02:19 PM - Alain Prouté

Providing a default value for $T is not possible, because $T may represent any type including an empty type. Such a default value will lead to inconsistencies.

Anyway, returning a default value for force_nth is also not good, because a default value is not the same as the n-th element of a list even if this element does not exist.

Of course, 'alert' is not in the philosophy of Anubis. However, it is legitimate as the body of a case when it is sure that this case will never happen. The problem with Anubis 1 is that the compiler has no way of ensuring that a case will never be used in a conditional.

There is only one correct solution to this problem. It consists in giving to the compiler the tools for ensuring that a case will never happen. This is based on statements and proofs, and will exist only in Anubis 2. There will be a keyword like 'can_never_happen', and the compiler will ask for a proof of this fact.

For the time being (Anubis 1). The best is probably to remove 'alert', and to let the programmers find solutions to their problems without alert. This is possible, but requires a better discipline.

Consequently, I suggest that everyone tries to remove all alerts from one's programs.

#2 - 09/23/2007 02:23 PM - Alain Prouté

- Status changed from New to Assigned

The first thing to do is to remove all alerts from the library itself. I must probably do it myself in order to demonstrate that this is possible without creating too much burden.

#3 - 09/23/2007 02:33 PM - Anonymous

Providing a default value is always possible... if this is the developper itself who provides it. This is what I mean.

I think the function *force_nth()* should be written like this :

```
   public define $T
     force_nth
       (
         Int32 n,
         List($T) l,
         $T default
       ) =
     if l is
       {
         [ ] then default,
         [h . t] then
           if n = 0
           then h
           else force_nth(n-1,t)
       }.
```

This is a lot more *anubisian* I think and this function is still simple and usefull...

#4 - 09/23/2007 02:35 PM - Anonymous

Oups... missing the recursive call :

```
   public define $T
     force_nth
       (
         Int32 n,
         List($T) l,
         $T default
       ) =
     if l is
       {
         [ ] then default,
         [h . t] then
           if n = 0
           then h
           else force_nth(n-1, t, default)
       }.
```

#5 - 09/23/2007 02:55 PM - Alain Prouté

Okay, you can do like this. So, this confirms that 'alert' should simply be removed.

Nevertheless, I suggest to keep 'alert' under another name like 'todo(...)' with a String message argument as a development tool. Also, if 'todo()' is encountered, the compiler should produce only a 'non release' version (notion to be defined...).

#6 - 10/18/2007 06:00 AM - Anonymous

The new force_nth() function with default parameter has been added.

But the old one is still present for compatibility. I think this bug have to be keep opened until we made all changes.

#7 - 10/26/2007 02:35 PM - Anonymous

- Status changed from Assigned to Closed

- Resolution set to fixed

Deprecated version of *force_nth()* has been removed, et the libraries code updated (tested with *compile_all.anubis*).