

## ***Gem #16: Прагма No\_Return***

**Автор: Bob Duff, AdaCore**

Краткое содержание: Gem Ада #16 - Прагму No\_Return можно использовать для пометки процедур, в которых невозможно обычным способом задать возвращение значения.

**Давайте начнем...**

Did he ever return,  
No he never returned  
And his fate is still unlearn'd  
He may ride forever  
'neath the streets of Boston  
He's the man who never returned.  
– from "Charlie on the M.T.A."  
– by The Kingston Trio

Он когда-нибудь возвращался,  
Нет, он никогда не возвращался  
И судьба его еще не  
Он может ездить вечно  
На улицах Бостона  
Он тот, кто никогда не возвращался.  
- от "Charlie on the M.T.A."  
- Трио Кингстон

Иногда полезно иметь процедуру, которая никогда не возвращается нормально, где «обычно» означает достижение «конца» или выполнение «возврата». Например, процедура может форматировать сообщение об ошибке на основе его параметров, отправлять сообщение в файл журнала и затем безоговорочно поднимать исключение. Другие способы избежать возврата обычно - это цикл навсегда, и ждать входного барьера, который никогда не станет True.

Такие процедуры необычны, и поэтому имеют право быть задокументированными. Комментарий работает, но прагма, No\_Return лучше (**pragma No\_Return**), потому что компилятор удостоверится, что процедура, на самом деле, не возвращается:

```
procedure Log_Error (...);  
pragma No_Return (Log_Error); -- Mark it as a non-returning procedure.  
  
procedure Log_Error (...) is  
begin  
  Put_Line (...);  
  raise Some_Error;  
end Log_Error;
```

**Pragma No\_Return** в Ada 2005 была вдохновлена определяемой реализацией прагмой No\_Return, которая существует в GNAT в течение некоторого времени. Теперь это стандартная особенность языка Ады.

В соответствии Ada RM, если процедура невозврата достигает “конца”, выполняется `raise Program_Error`. Но это - просто легкая опасность; то, что мы действительно хотим, является проверкой времени компиляции. На самом деле, если будет какой-либо путь потока управления, который может достигнуть “конца”, предположительно, процедуры невозврата, GNAT даст предупреждение. Так же всегда Вы можете сказать GNAT рассматривать предупреждения как ошибки. Эти предупреждения обязательно консервативны, таким образом, Вы, возможно, иногда должны были бы использовать `pragma Warnings (Off)`, чтобы препятствовать тому, чтобы компилятор “поднял ложную тревогу”.

See also paragraph 6.5.1(1.a/2) of the Annotated Ada Reference Manual, regarding `pragma No_Deposit`: <http://www.adaic.org/standards/05aarm/html/AA-6-5-1.html>

См. Также пункт 6.5.1 (1.a / 2) Справочного руководства по аннотированной Аде, касающийся прагмы `No_Deposit`: <http://www.adaic.org/standards/05aarm/html/AA-6-5-1.html>

**Связанный исходный код**

**Обсуждение...**