GcaCodeRun

Функция **GcaCodeRun** исполняет код, загруженный в ключ, из другого участка загруженного кода.

C

```
int GRD_API GcaCodeRun(
  HANDLE hGrd,
  DWORD dwAlgoName,
  DWORD dwP1,
  DWORD *pdwRet,
  DWORD dwDataFromDongleLng,
  void *pDataFromDongle,
  DWORD dwDataToDongleLng,
  void *pDataToDongle,
  void *pReserved
);
```

hGrd	Не используется.
dwAlgoName	Числовое имя алгоритма
dwP1	Параметр, передаваемый в подфункцию (аналог **main**) загружаемого кода.
pdwRet	Параметр, возвращаемый из подфункции (аналог **main**) загружаемого кода.
dwDataFromDongleLng	Размер блока данных, получаемых из ключа
pDataFromDongle	Указатель на буфер для блока данных, получаемых из ключа
dwDataToDongleLng	Размер блока данных передаваемых в ключ.
pDataToDongle	Указатель на буфер для блока данных, передаваемых в ключ
pReserved	Зарезервировано. Должно быть **NULL**.

Набор ошибок Guardant API

Функция **GcaCodeRun** аналогична GrdCodeRun из внешнего Guardant API, она служит для запуска кода, предварительно загруженного в ключ, из другого участка загруженного в ключ кода.

При использовании **GcaCodeRun**, области ОЗУ ***запускающего*** и ***запускаемого*** загружаемого кода не должны пересекаться или совпадать для предотвращения затирания общей памяти. Рекурсия запрещена. При попытке вызова **GcaCodeRun** в описанных выше ситуациях возвращается код ошибки GrdE_gcRamAccessViolation. Глубина таких вызовов равна 1. Таким образом, при попытке запуска загружаемого кода из другого загружаемого кода, запущенного, в свою очередь, с помощью **GcaCodeRun**, возвращается код ошибки GrdE_gcCallDepthOverflow.

В **main**-функцию пользовательского кода передается параметр *dwP1* и буфер данных *pDataToDongle*. По завершению работы **main**-функции пользовательского кода, она возвращает *pdwRet* и буфер данных *pDataToDongle*.

Время работы **main** функции пользовательского кода ограничено 3 сек по умолчанию, если иное не задано GcaSetTimeout.