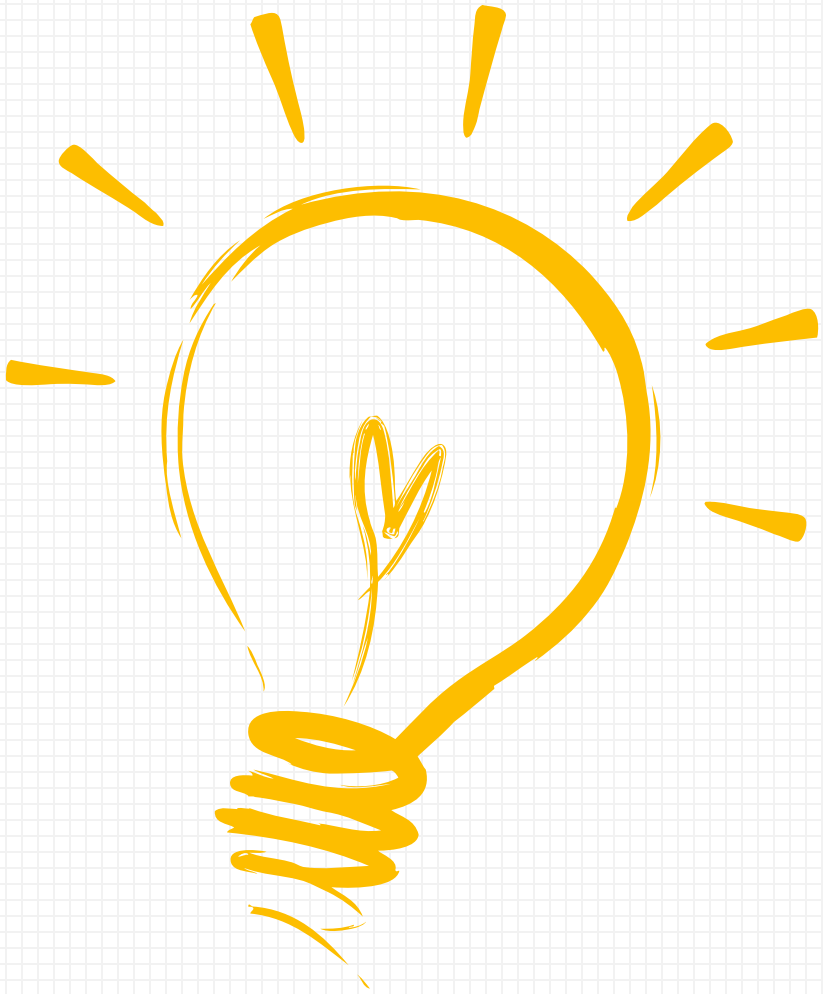
CUDA错误检查

CUDA并行编程系列课程

主讲：权双

CONTENTS



运行时API错误代码

★ 1、CUDA运行时API大多支持返回错误代码，返回值类型：cudaError_t

★ 2、运行时API成功执行，返回值为cudaSuccess

★ 3、运行时API返回的执行状态值是枚举变量

Values

cudaSuccess = 0

The API call returned with no errors. In the case of query calls, this also means that the operation being queried is complete (see cudaEventQuery() and cudaStreamQuery()).

cudaErrorInvalidValue = 1

This indicates that one or more of the parameters passed to the API call is not within an acceptable range of values.

cudaErrorMemoryAllocation = 2

The API call failed because it was unable to allocate enough memory to perform the requested operation.

cudaErrorInitializationError = 3

The API call failed because the CUDA driver and runtime could not be initialized.

cudaErrorCudartUnloading = 4

错误检查函数

★ 1、获取错误代码对应名称：cudaGetErrorName

★ 2、获取错误代码描述信息： cudaGetErrorString

__host____device__const char *cudaGetErrorName (cudaError_t error)

Returns the string representation of an error code enum name.

Parameters

error

- Error code to convert to string

Returns

`char*` pointer to a NULL-terminated string

Description

Returns a string containing the name of an error code in the enum. If the error code is not recognized, "unrecognized error code" is returned.

__host____device__const char *cudaGetErrorString (cudaError_t error)

Returns the description string for an error code.

Parameters

error

- Error code to convert to string

Returns

`char*` pointer to a NULL-terminated string

Description

Returns the description string for an error code. If the error code is not recognized, "unrecognized error code" is returned.

错误检查函数

★ 1、在调用CUDA运行时API时，调用ErrorCheck函数进行包装

★ 2、参数filename一般使用__FILE__; 参数lineNumber一般使用__LINE__

★ 3、错误函数返回运行时API调用的错误代码

```
34
35  cudaError_t ErrorCheck(cudaError_t error_code, const char* filename, int lineNumber)
36  {
37      if (error_code != cudaSuccess)
38      {
39          printf("CUDA error:\r\ncode=%d, name=%s, description=%s\r\nfile=%s, line=%d\r\n",
40                  error_code, cudaGetErrorName(error_code), cudaGetErrorString(error_code), filename, lineNumber);
41          return error_code;
42      }
43      return error_code;
44  }
45
46
```

检查核函数

★ 错误检测函数问题：不能捕捉调用核函数的相关错误

★ 捕捉调用核函数可能发生错误的方法：

ErrorCheck(cudaGetLastError(), __FILE__, __LINE__);

ErrorCheck(cudaDeviceSynchronize(), __FILE__, __LINE__);

★ 核函数定义：

__global__ void kernel_function(argument arg);

__host____device__cudaError_t cudaGetLastError (void)

Returns the last error from a runtime call.

Returns

cudaSuccess, cudaErrorMissingConfiguration, cudaErrorMemoryAllocation, cudaErrorInitializationError, cudaErrorLaunchFailure, cudaErrorLaunchTimeout, cudaErrorLaunchOutOfResources, cudaErrorInvalidDeviceFunction, cudaErrorInvalidConfiguration, cudaErrorInvalidDevice, cudaErrorInvalidValue, cudaErrorInvalidPitchValue, cudaErrorInvalidSymbol, cudaErrorUnmapBufferObjectFailed, cudaErrorInvalidDevicePointer, cudaErrorInvalidTexture, cudaErrorInvalidTextureBinding, cudaErrorInvalidChannelDescriptor, cudaErrorInvalidMemcpyDirection, cudaErrorInvalidFilterSetting, cudaErrorInvalidNormSetting, cudaErrorUnknown, cudaErrorInvalidResourceHandle, cudaErrorInsufficientDriver, cudaErrorNoDevice, cudaErrorSetOnActiveProcess, cudaErrorStartupFailure, cudaErrorInvalidPtx, cudaErrorUnsupportedPtxVersion, cudaErrorNoKernelImageForDevice, cudaErrorJitCompilerNotFound, cudaErrorJitCompilationDisabled

Description

Returns the last error that has been produced by any of the runtime calls in the same instance of the CUDA Runtime library in the host thread and resets it to cudaSuccess.

Note: Multiple instances of the CUDA Runtime library can be present in an application when using a library that statically links the CUDA Runtime.

THANKS

谢谢聆听