

# HƯỚNG DẪN CÀI ĐẶT CUDA

## TRÊN MÁY KHÔNG HỖ TRỢ CUDA

### 1. Hướng dẫn cài đặt

💡 Download bộ cài đặt lập trình CUDA tại địa chỉ:

[http://www.nvidia.com/object/cuda\\_get.html](http://www.nvidia.com/object/cuda_get.html)

Trong trang download của CUDA thực hiện:

- Chọn phiên bản mới nhất là CUDA 2.2
- Chọn hệ điều hành phù hợp (Vista/XP)
- Tiến hành download:
  - CUDA Toolkit
  - CUDA SDK

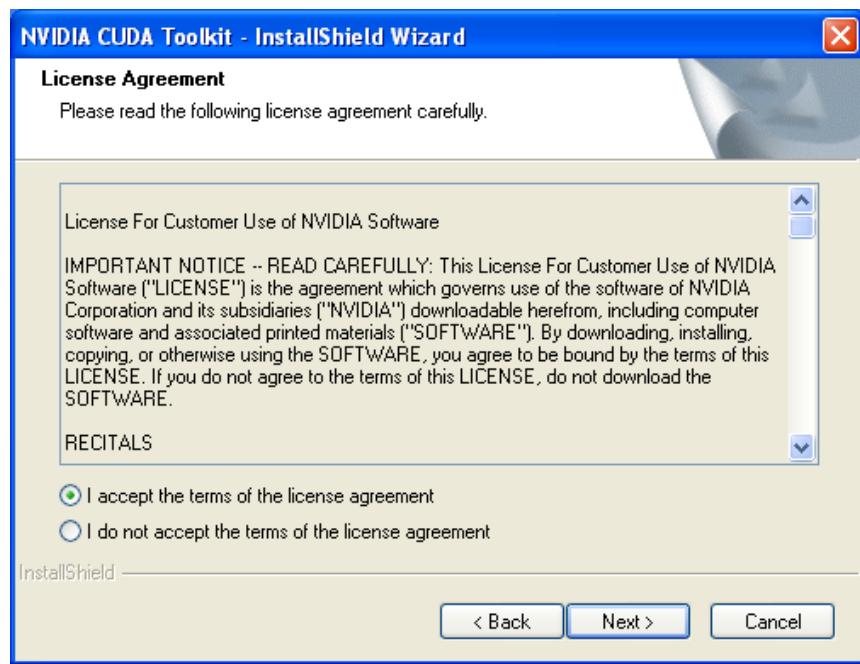
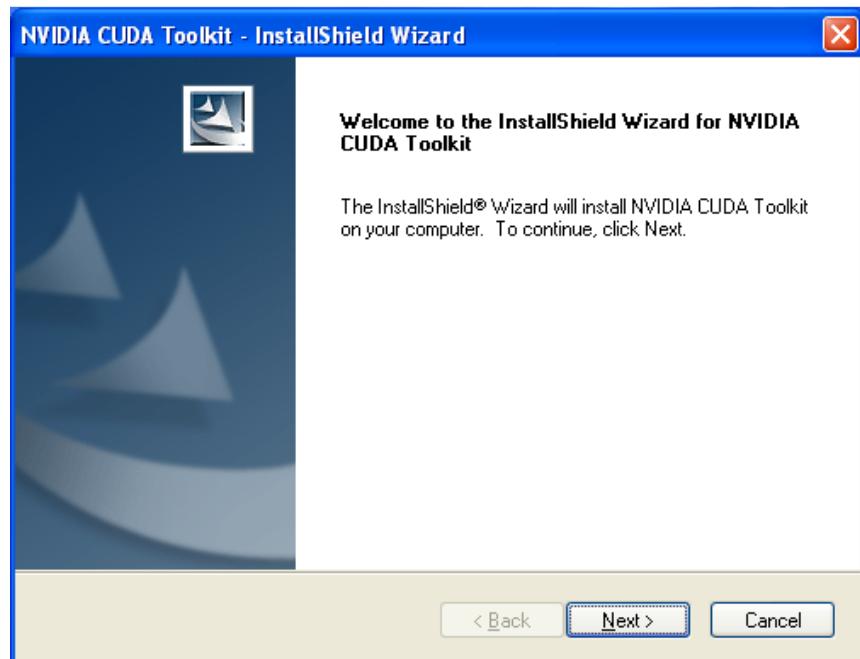
*Lưu ý:* không cần download Driver vì máy không có card đồ họa của NVIDIA và ta cũng sẽ không cần cài driver.

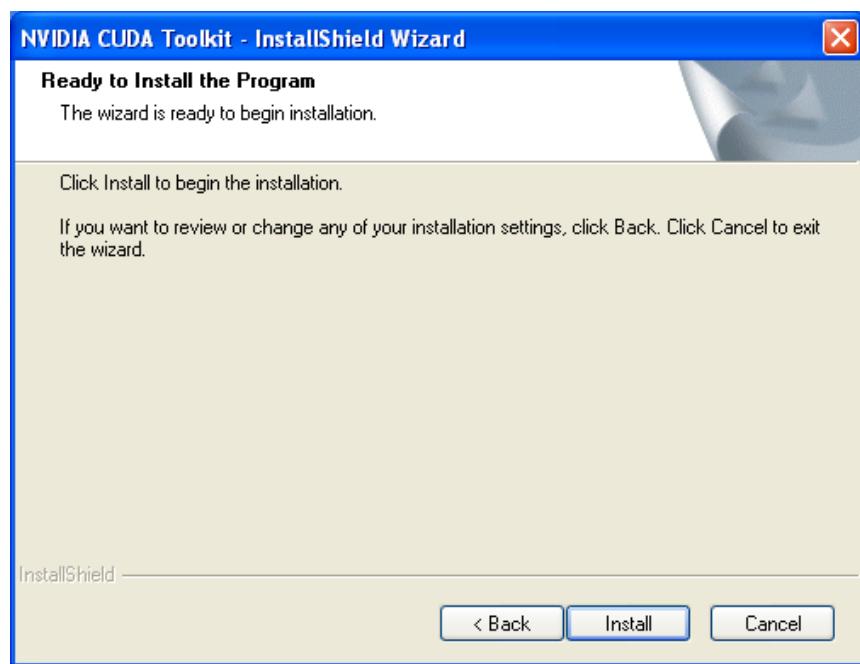
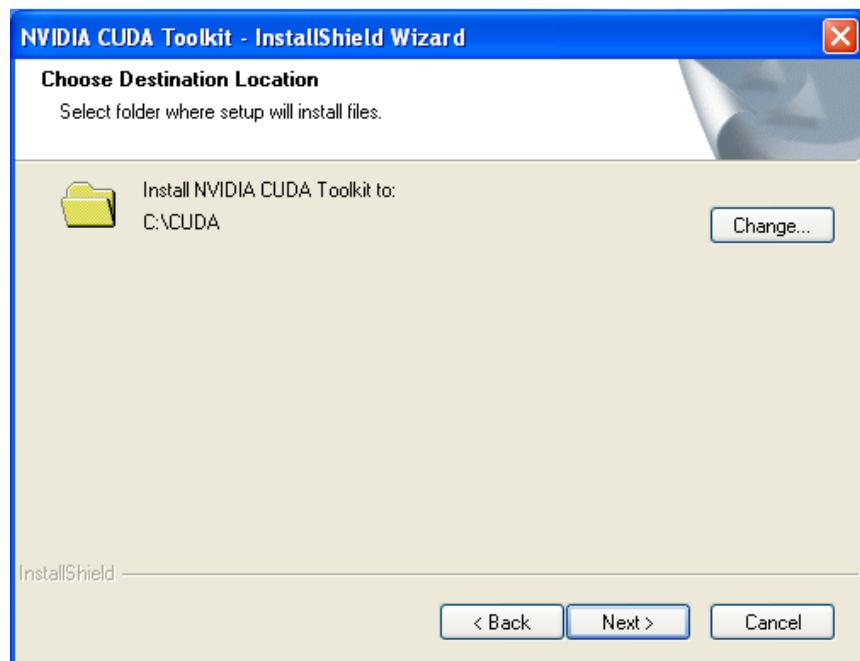
💡 **Tiến hành cài đặt**

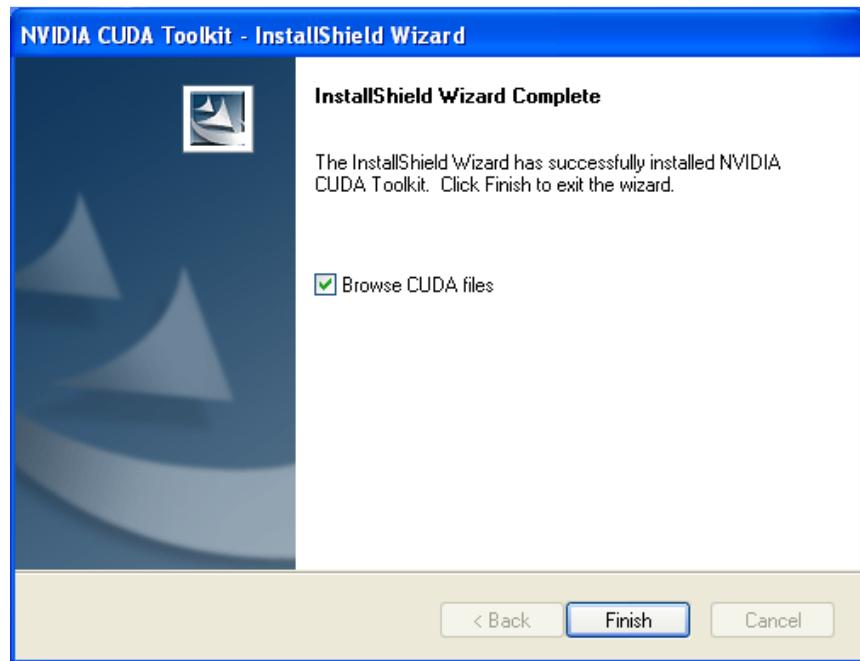
Sau khi download thành công, chúng ta sẽ có **2** tập tin .exe để cài đặt. Việc cài đặt bộ lập trình CUDA phải theo **đúng** thứ tự sau: cài **CUDA Toolkit** sau đó là cài **CUDA SDK**.

#### a. Cài đặt CUDA Toolkit

Chạy tập tin .exe để cài đặt

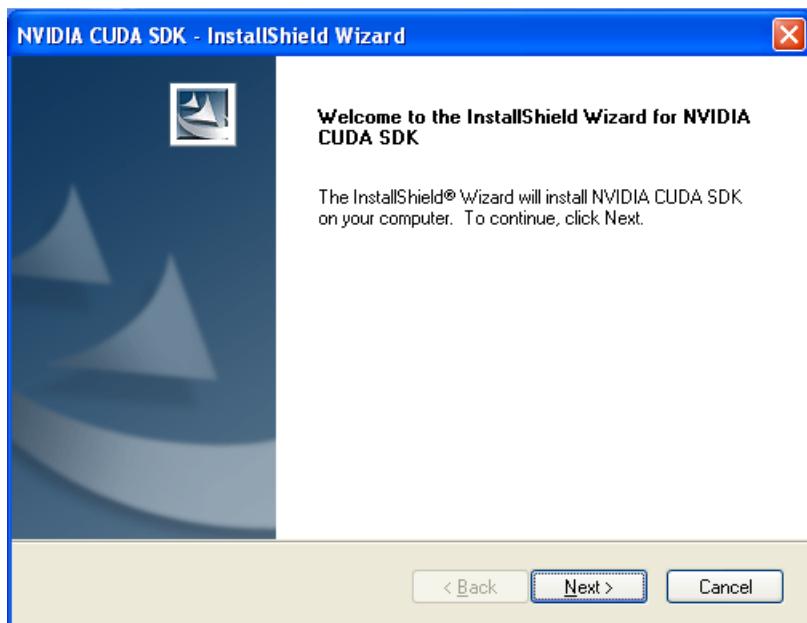


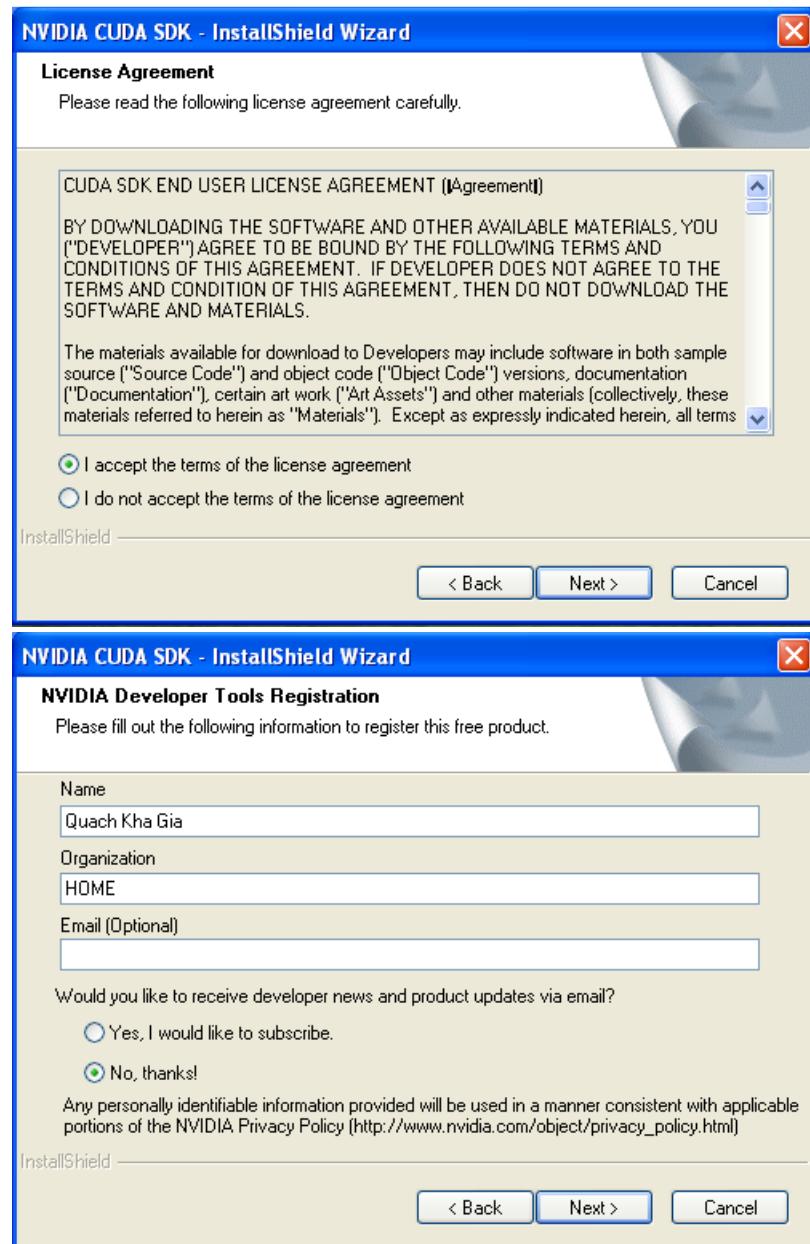


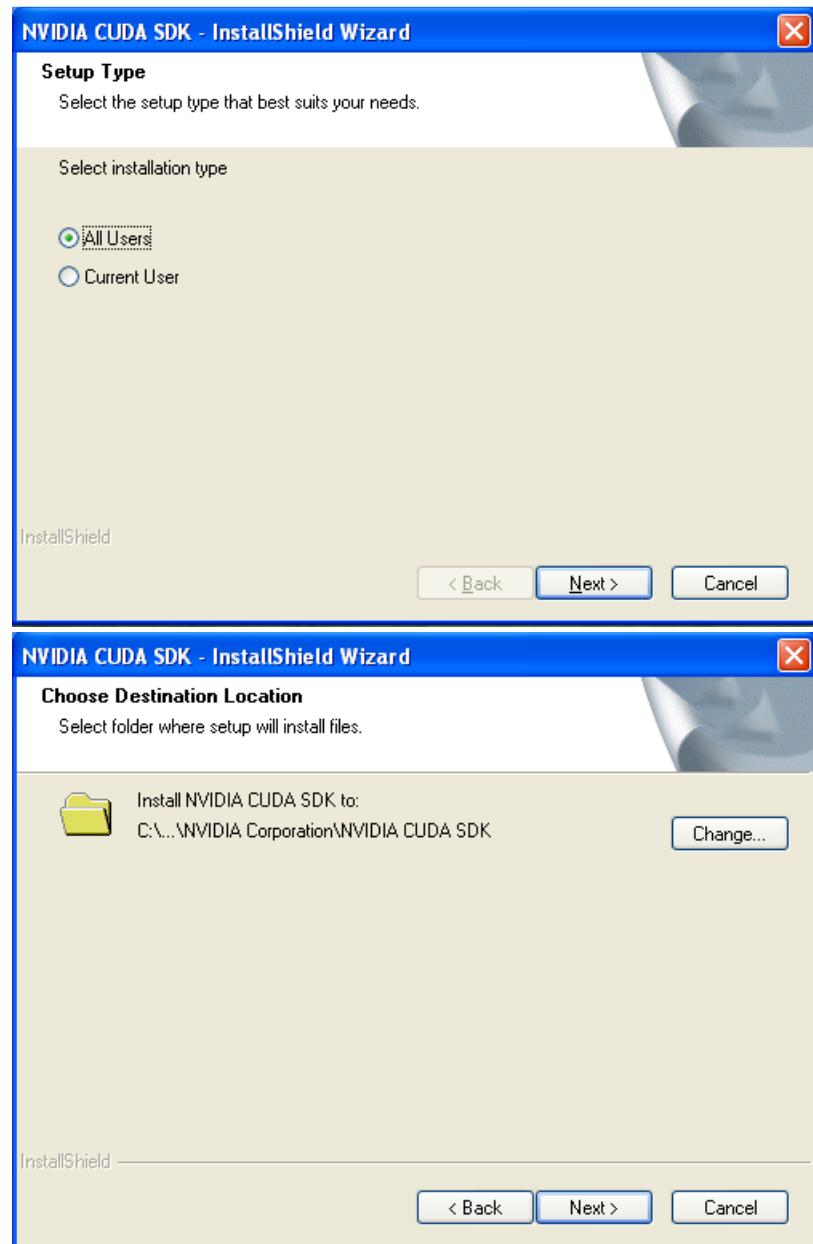


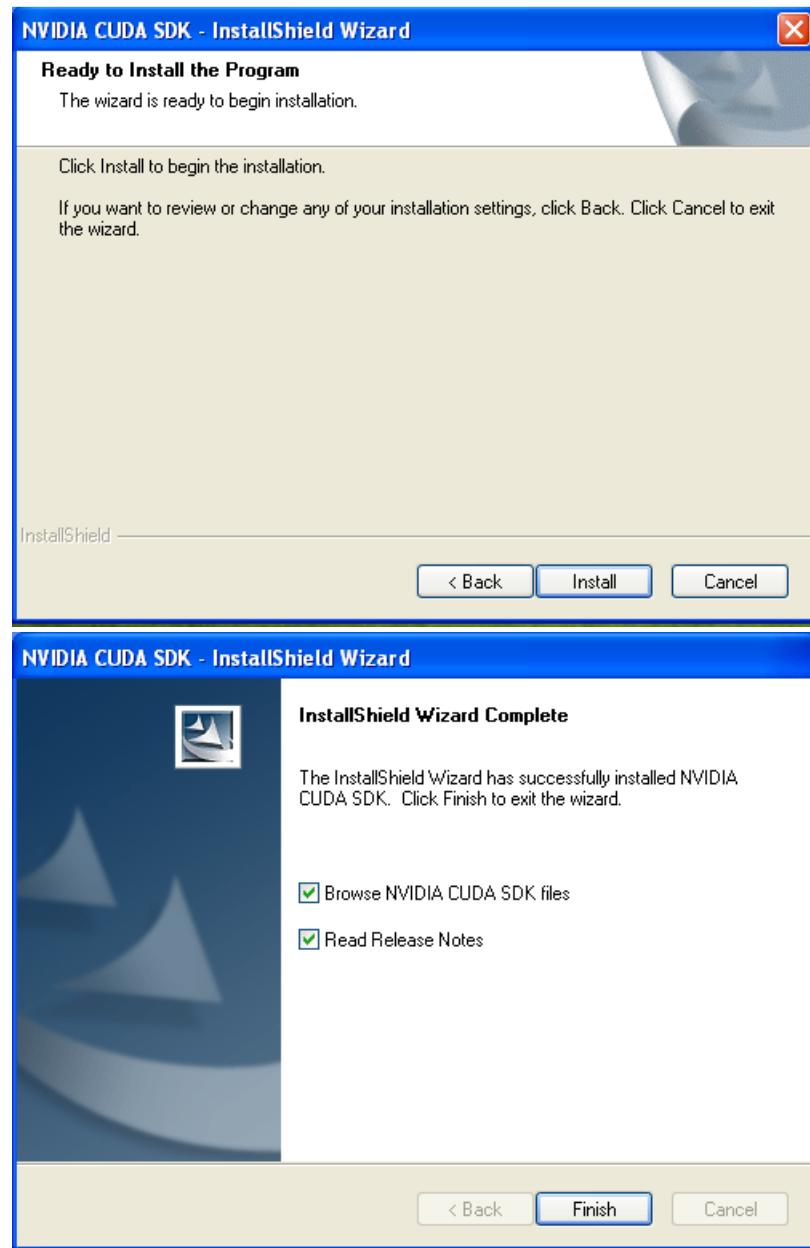
**b. Cài đặt CUDA SDK**

Chạy tập tin .exe để cài đặt









**Lưu ý:** Sau khi cài đặt xong CUDA nên restart lại máy để bước tiếp theo có thể tiến hành thuận lợi

## 2. Biên dịch CUDA trong Visual Studio 2005

Mở những project mẫu (Release.sln) kèm theo bộ CUDA SDK trong thư mục C:\Program Files\NVIDIA Corporation\NVIDIA CUDA SDK\projects\  
Chọn một project trong Solution Explorer rồi vô

- Menu *Build* → *Build <Tên project>*

- Menu Project → Set as StartUp Project

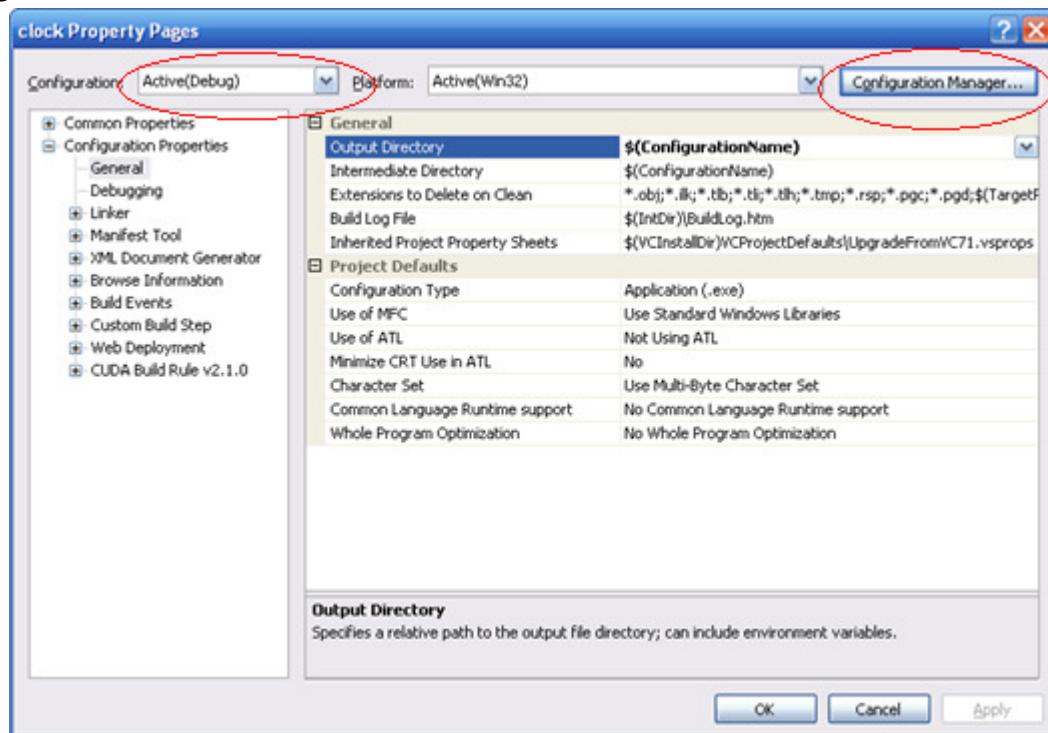
Nhấn *Ctrl + F5* để chạy thử

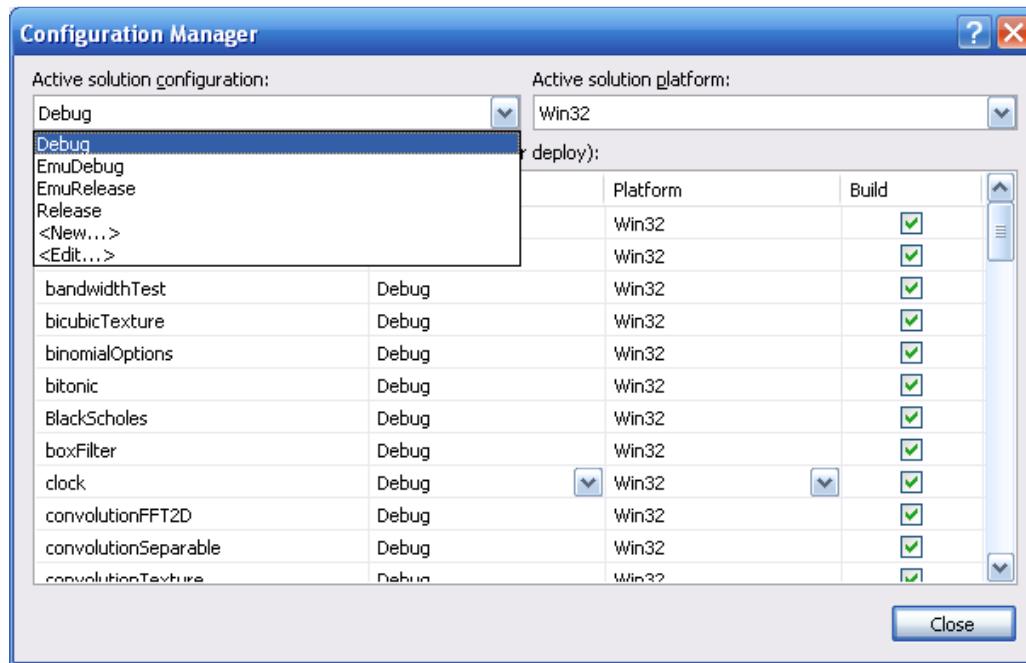
Kết quả chạy cho project “Transpose”. Đây là kết quả chạy không thành công vì chương trình đã báo lỗi Runtime “*feature is not yet implemented*”. Đối với những máy không hỗ trợ CUDA ta phải thiết lập lại chế độ biên dịch (sử dụng chế độ giả lập) để có thể chạy được.

Một số project cần có các thư viện khác để có thể biên dịch và chạy được (vd: DirectX SDK, GLUT, ...)

```
C:\WINDOWS\system32\cmd.exe
cudaSafeCall() Runtime API error in file <clock.cu>, line 92 : feature is not yet implemented.
Press any key to continue . . .
```

Khi xem *Properties* của Project thì ta thấy chế độ biên dịch mặc định là **Debug**. Để chỉnh lại chế độ biên dịch giả lập nhấn “*Configuration Manager*” Khi đó sẽ hiện ra bảng như hình



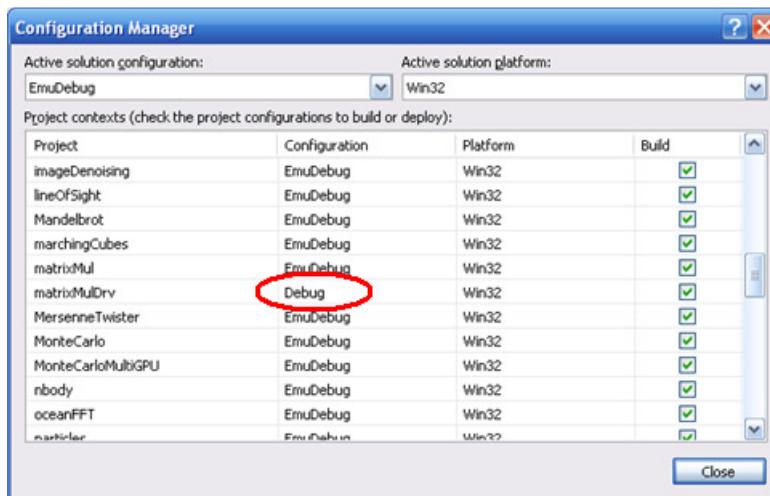


Trong list các chế độ ta thấy có nhiều mục lựa chọn: **Debug**, **Release**, **EmuDebug** và **EmuRelease**

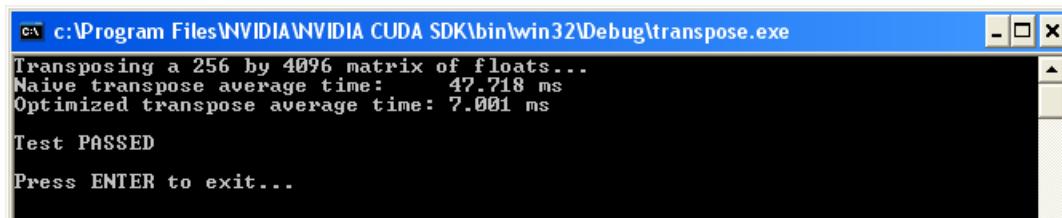
Chữ “Emu” ở đầu thẻ hiện đây là chế độ giả lập. Như vậy CUDA SDK hỗ trợ hai chế độ **giả lập** (**EmuDebug** và **EmuRelease**) tương ứng với hai chế độ **thật** **Debug** và **Release** (chỉ chạy được trên máy có card NVIDIA).

Sau khi chọn EmuDebug toàn bộ các project sẽ được biên dịch ở chế độ giả lập debug.

**Lưu ý:** Có một số project không hỗ trợ build ở chế độ giả lập. Mặc dù các project khác đã chuyển sang EmuDebug nhưng có một số project lại không chuyển được (xem hình minh sau)



Kết quả chạy trên máy có card NVIDIA

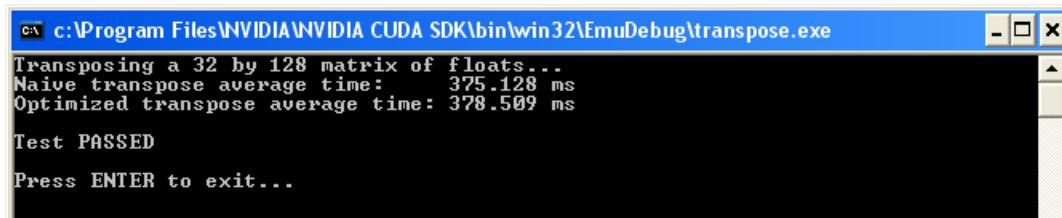


```
c:\Program Files\NVIDIA\NVIDIA CUDA SDK\bin\win32\Debug\transpose.exe
Transposing a 256 by 4096 matrix of floats...
Naive transpose average time:    47.718 ms
Optimized transpose average time: 7.001 ms

Test PASSED

Press ENTER to exit...
```

Kết quả chạy giả lập trên máy không có card NVIDIA



```
c:\Program Files\NVIDIA\NVIDIA CUDA SDK\bin\win32\EmuDebug\transpose.exe
Transposing a 32 by 128 matrix of floats...
Naive transpose average time:    375.128 ms
Optimized transpose average time: 378.509 ms

Test PASSED

Press ENTER to exit...
```

Qua đó thấy rõ tác dụng của việc chạy trên thiết bị thực. Thiết bị thực cho kết quả chạy nhanh hơn chạy giả lập đến **50** lần.

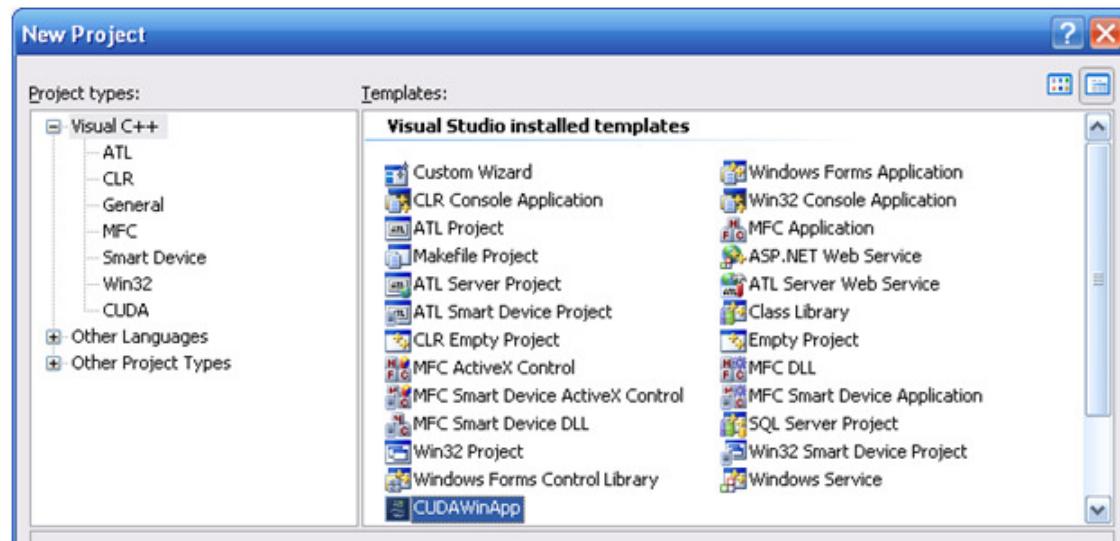
### 3. Tạo project CUDA trong Visual Studio 2005

Có hai cách để thực hiện

#### a. Sử dụng CUDA VS Wizard

Download file cài đặt từ trang <http://sourceforge.net/projects/cudavswizard/>

Sau khi cài đặt xong, mở VS 2005 sẽ thấy trong dialog *New Project -> Project Type* (Visual C++) có thêm một mục mới **CUDAWinApp**.



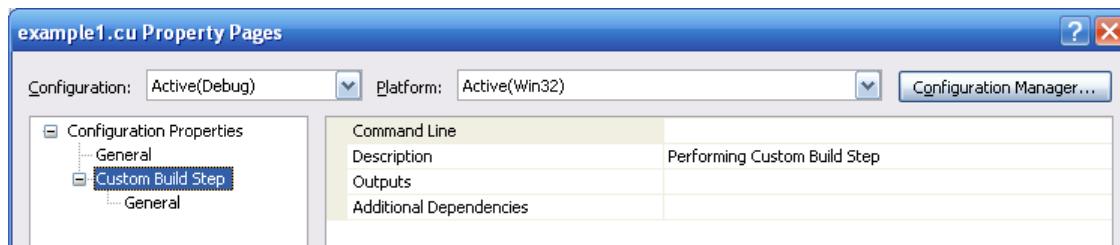
### b. Tự thiết lập

Trong thiết lập mặc định của VC++ không hỗ trợ biên dịch file .cu, ta cần chỉ cho VC++ biết cách để biên dịch file .cu này. Thực hiện điều này bằng cách dùng *Custom Build Step*. Nhấn chuột phải trên file .cu đã add vào project và chọn *Properties*. Trong cửa sổ *Property Pages* hiện ra, thiết lập *Custom Build Step* như sau:

*Configuration Properties* → *Custom Build Step* → *General*:

*Command Line* =

```
“$(CUDA_BIN_PATH)\nvcc.exe” -ccbin “$(VCInstallDir)bin” -c -D_DEBUG -DWIN32 -deviceemu -D_CONSOLE -D_MBCS -Xcompiler /EHsc,/W3,/nologo,/Wp64,/Od,/Zi,/MTd -I’$(CUDA_INC_PATH)’ -I./ -o $(ConfigurationName)\example1.obj example1.cu
```



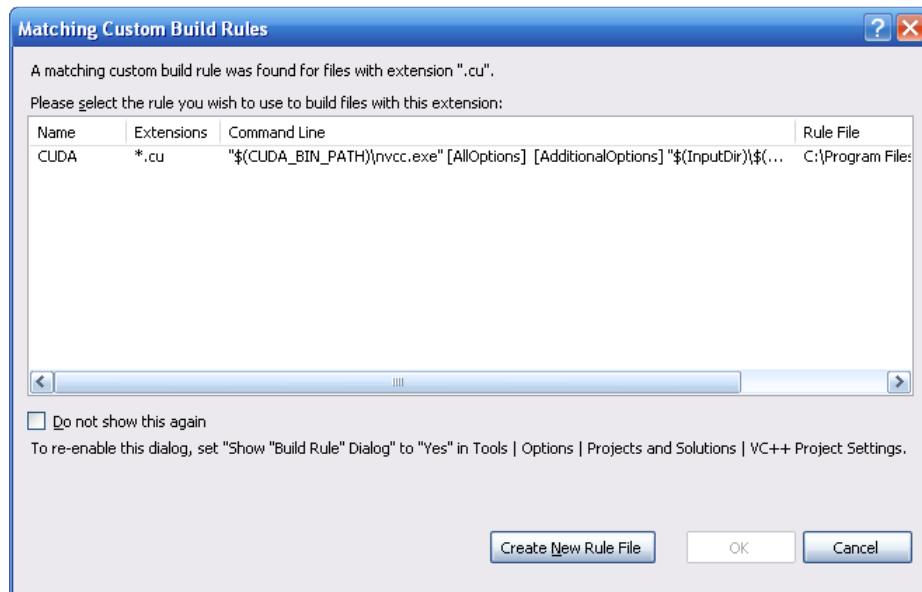
Ý nghĩa của các thành phần trong *Command Line* là gì, ta tách ra từng phần để xem xét.

**“\$(CUDA\_BIN\_PATH)\nvcc.exe”**: Đường dẫn tới trình biên dịch nvcc.  
**-ccbin “\$(VCInstallDir)bin”**: Đường dẫn tới trình biên dịch của VC++.  
**-c**: Quá trình biên dịch được thực hiện đến khi phát sinh ra file object (.obj).  
**-D\_DEBUG -DWIN32 -D\_CONSOLE -D\_MBCS**: tham số cho nvcc  
**-deviceemu**: biên dịch ở chế độ giả lập  
**-Xcompiler /EHsc,/W3,/nologo,/Wp64,/Od,/Zi,/MTd**: Những tham số cho trình biên dịch VC++  
**-I “\$(CUDA\_INC\_PATH)”**: tìm trong thư mục include của CUDA Toolkit những file .h được include.  
**-I/**: tìm trong thư mục hiện hành những file .h được include.  
**-o \$(ConfigurationName)\example1.obj**: Tên và đường dẫn của file kết quả biên dịch.  
**example1.cu**: Tên file mã nguồn.

Bên cạnh đó ta phải thiết lập đường dẫn cho file kết xuất ra.

*Configuration Properties → Custom Build Step → General:*  
Outputs = \$(ConfigurationName)\example1.obj

Nếu đã cài đặt CUDA VS Wizard thì khi add một file .cu có sẵn vào project thì sẽ hiện ra cửa sổ như sau:



Chọn dòng CUDA và nhấn OK, khi đó file .cu mới add vào sẽ được tự động thiết lập luôn, không cần thực hiện các bước thiết lập ở trên nữa.

Sau khi thiết lập *properties* của file .cu, ta cần chỉnh thêm *properties* của *project*.

*Configuration Properties* → *C/C++* → *General*:

*Additional Include Directories* = \$(CUDA\_INC\_PATH);"  
\$(NVSDKCUDA\_ROOT)common\lib"

*Configuration Properties* → *C/C++* → *General*:

*Debug Information Format* = Program Database (/Zi)

*Configuration Properties* → *C/C++* → *Code Generation*:

*Runtime Library* = Multi-threaded Debug (/MTd)

*Configuration Properties* → *Linker* → *General*:

*Enable incremental linking* = No (/INCREMENTAL:NO)

*Configuration Properties* → *Linker* → *General*:

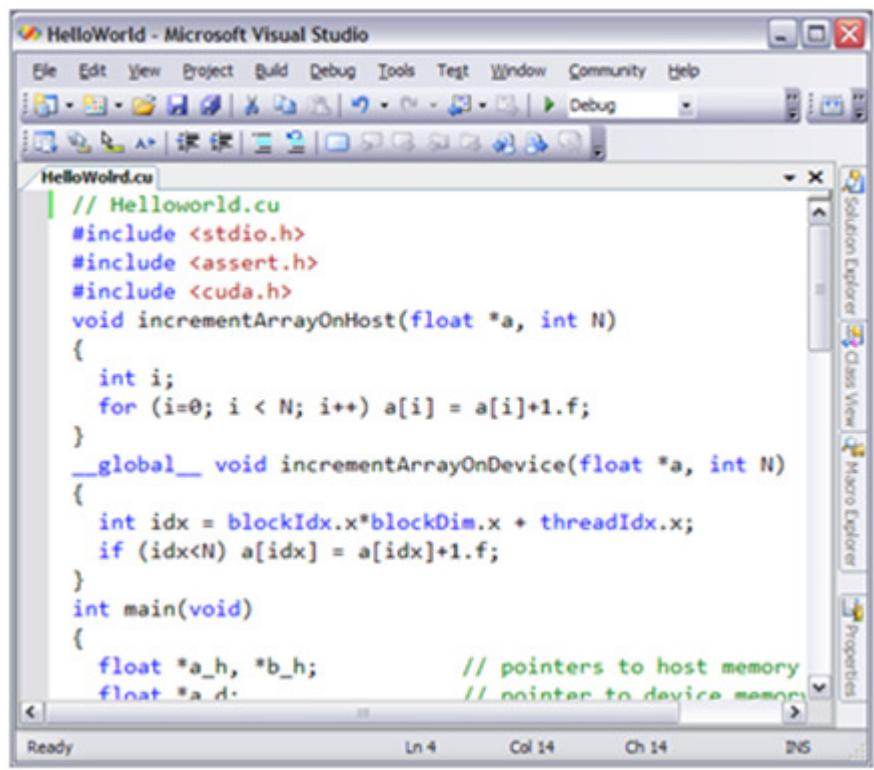
*Additional Library Directories* = "\$(CUDA\_LIB\_PATH)";"  
\$(NVSDKCUDA\_ROOT)common\lib"

*Configuration Properties → Linker → Input:  
Additional Dependencies = cudart.lib cutil32D.lib*

*Configuration Properties → Linker → Optimization:  
Enable COMDAT folding = Do Not Remove Redundant COMDATs  
(/OPT:NOICF)*

#### 4. Highlight cú pháp CUDA trong Visual Studio 2005

Khi làm việc trên những file code của CUDA (có phần mở rộng .cu) thì VS 2005 xem như một file text bình thường và không có highlight những từ khóa như với file .cpp của C++. Tuy nhiên, ta có thể làm cho VS 2005 hỗ trợ file .cu này một cách dễ dàng và đạt được hiệu quả như trong hình sau.

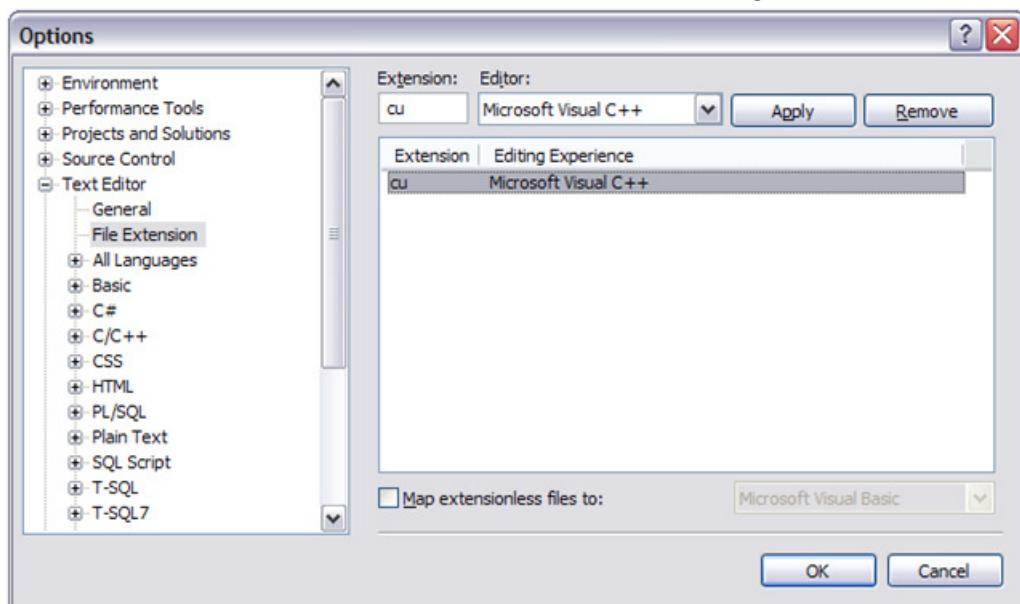


```
// HelloWorld.cu
#include <stdio.h>
#include <assert.h>
#include <cuda.h>
void incrementArrayOnHost(float *a, int N)
{
    int i;
    for (i=0; i < N; i++) a[i] = a[i]+1.f;
}
__global__ void incrementArrayOnDevice(float *a, int N)
{
    int idx = blockIdx.x*blockDim.x + threadIdx.x;
    if (idx<N) a[idx] = a[idx]+1.f;
}
int main(void)
{
    float *a_h, *b_h;           // pointers to host memory
    float *a_d;                // pointer to device memory
}
```

#### Cách bước thực hiện:

1. Vào trong thư mục “Microsoft Visual Studio 8\Common7\IDE” nằm trong thư mục chứa VS 2005 (VD: “C:\Program Files”)

2. Mở file “usertype.dat” (nếu có) trong thư mục này bằng notepad. Nếu chưa có, tạo một file mới đặt tên giống như vậy.
3. Vào trong thư mục cài đặt CUDA SDK “%Program Files%\NVIDIA Corporation\NVIDIA CUDA SDK\doc\syntax\_highlighting\visual\_studio\_8”
4. Thêm nội dung của file “usertype.dat” trong thư mục trên vào file “usertype.dat” đang mở từ “Microsoft Visual Studio 8\Common7\IDE”
5. Lưu lại file này.
6. Mở Visual Studio và chọn menu *Tools → Options*
7. Chọn mục *Text Editor → File Extension*. Nhập “cu” trong *Extension* và chọn *Editor* là Microsoft Visual C++, sau đó nhấn *Add* như trong hình sau.



8. Khởi động lại VS. Sẽ được kết quả highlight.