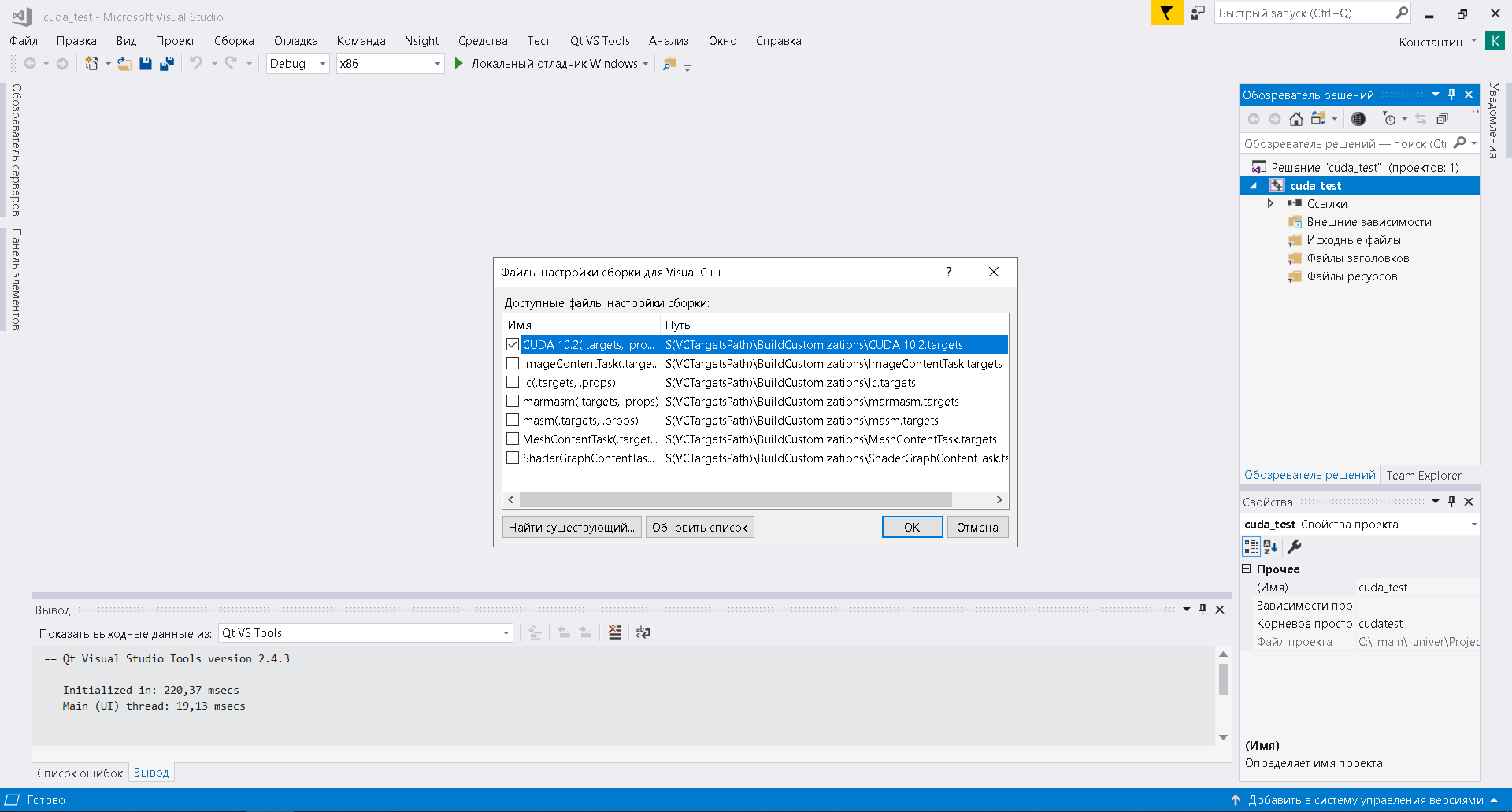
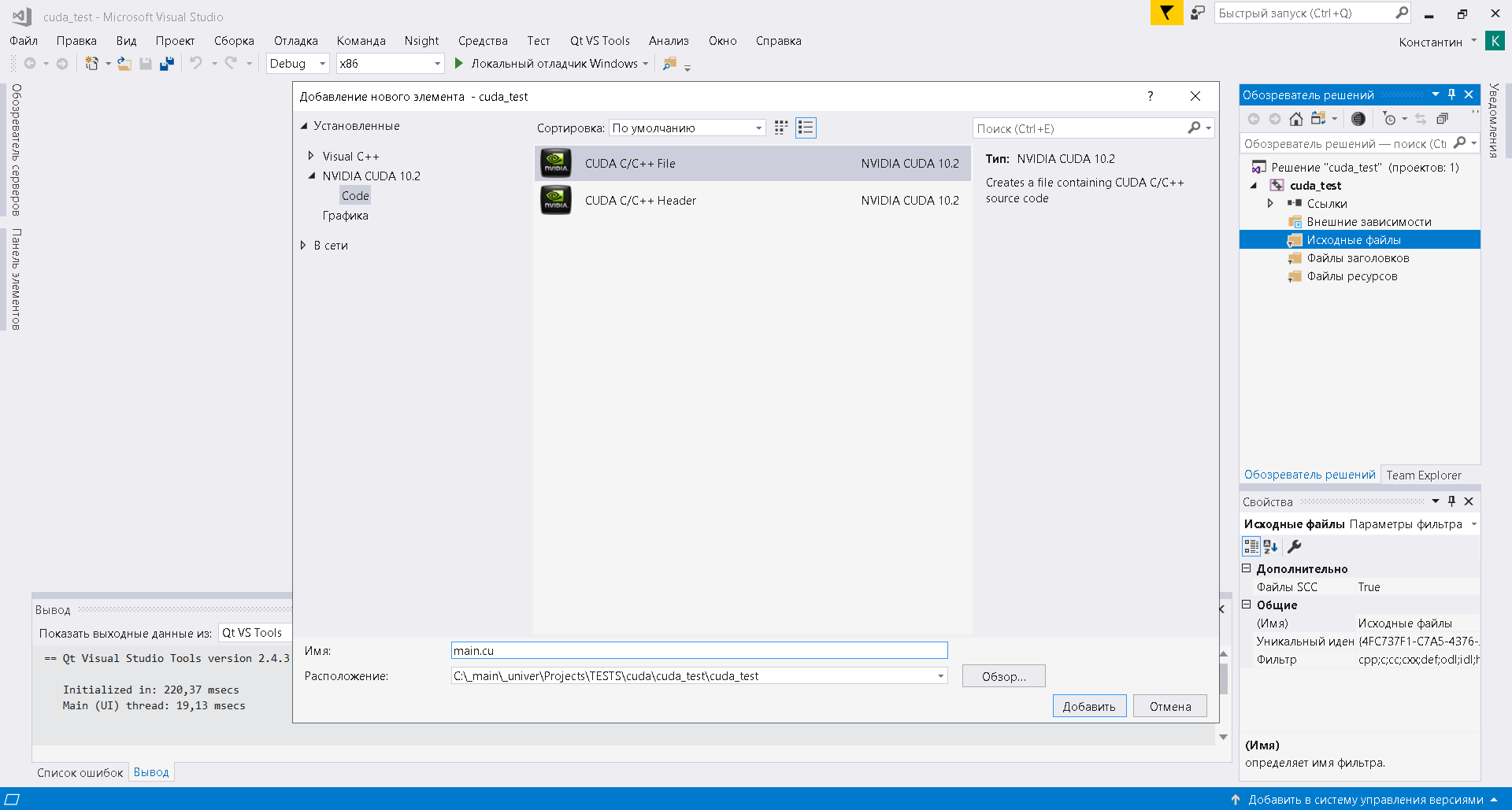
1.Создаём пустой С++ проект.

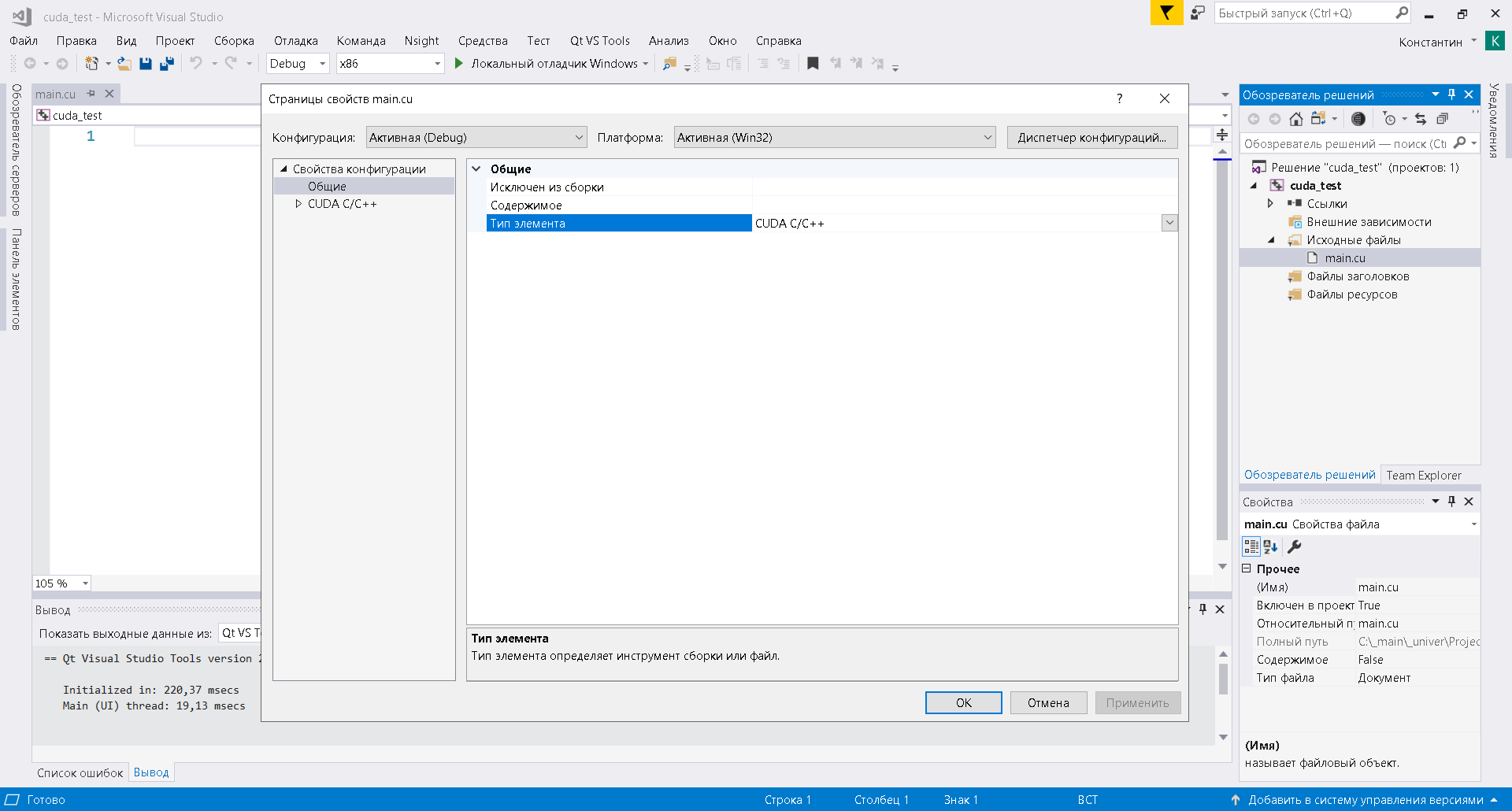
2.Правой по проекту в обозревателе решений -> зависимости сборки -> настройки сборки -> галочка на нужной версии CUDA:



3.Исходные файлы -> добавить -> создать элемент -> NVIDIA CUDA -> Code -> CUDA C/C++ file пишем имя main.cu:

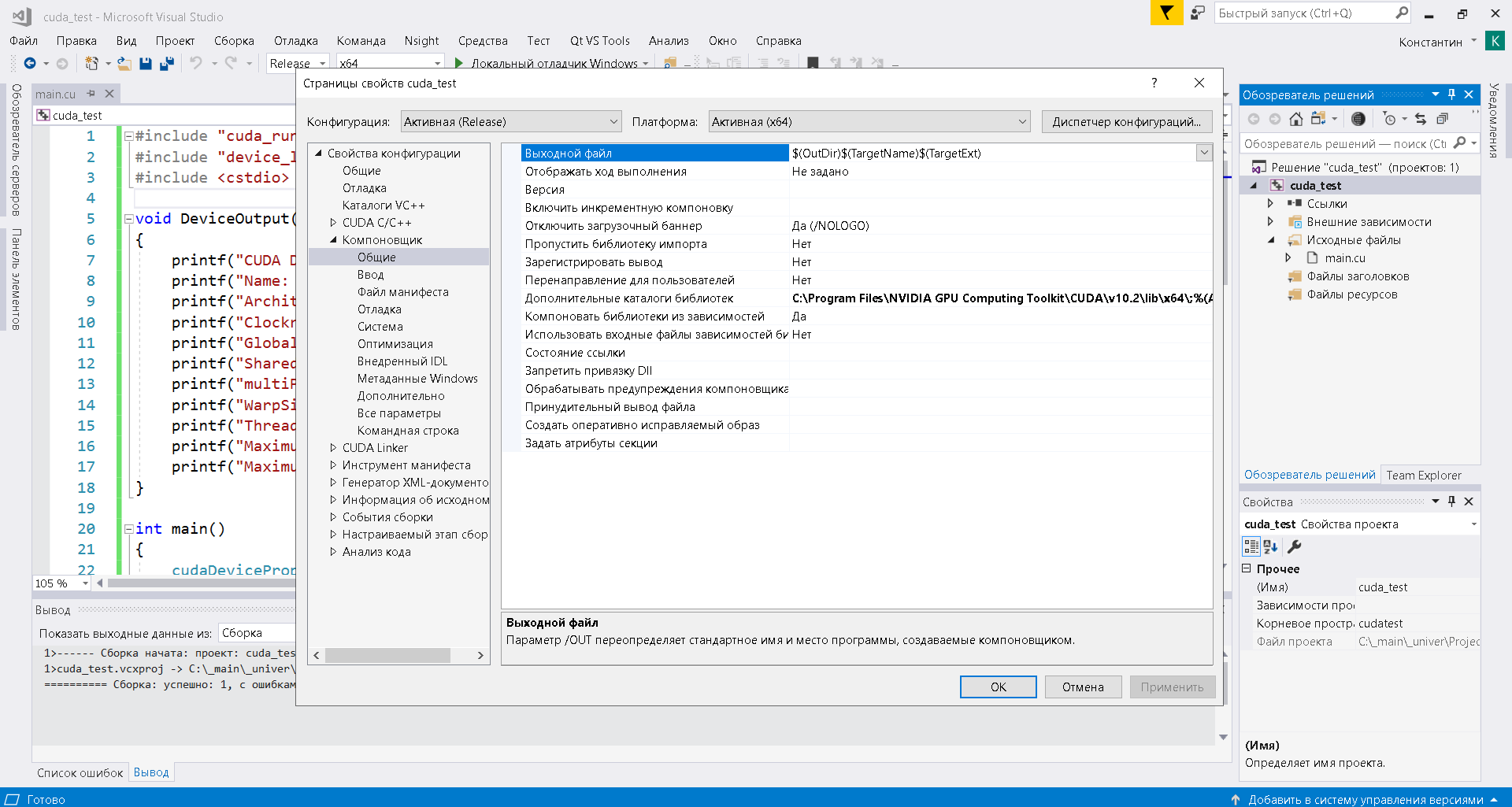
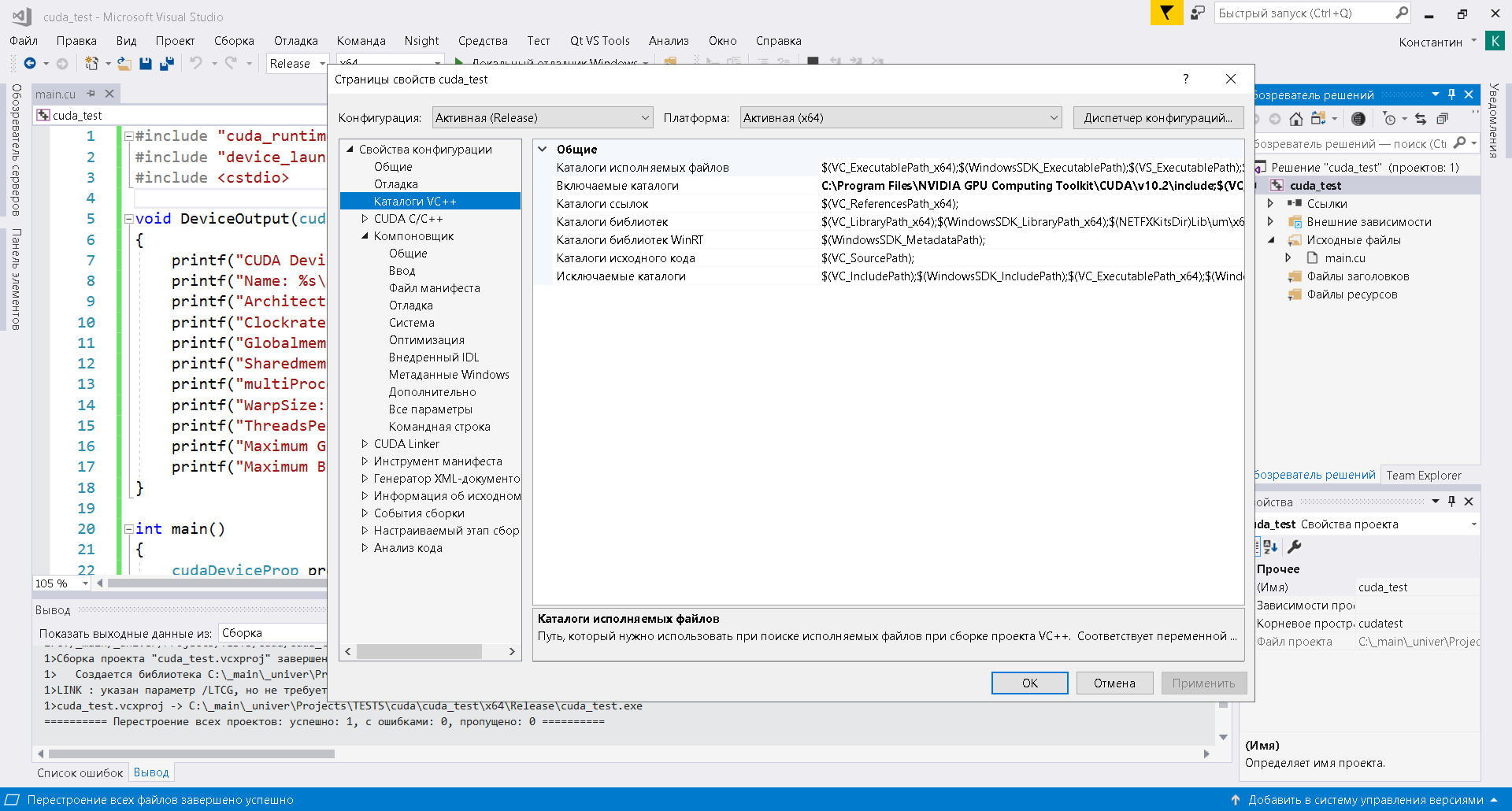


4.В свойствах файла main.cu проверяем, что стоит тип элемента CUDA C/C++, это означает, что его обработкой будет заниматься не компилятор VS, а компилятор Nvidia (nvcc.exe):

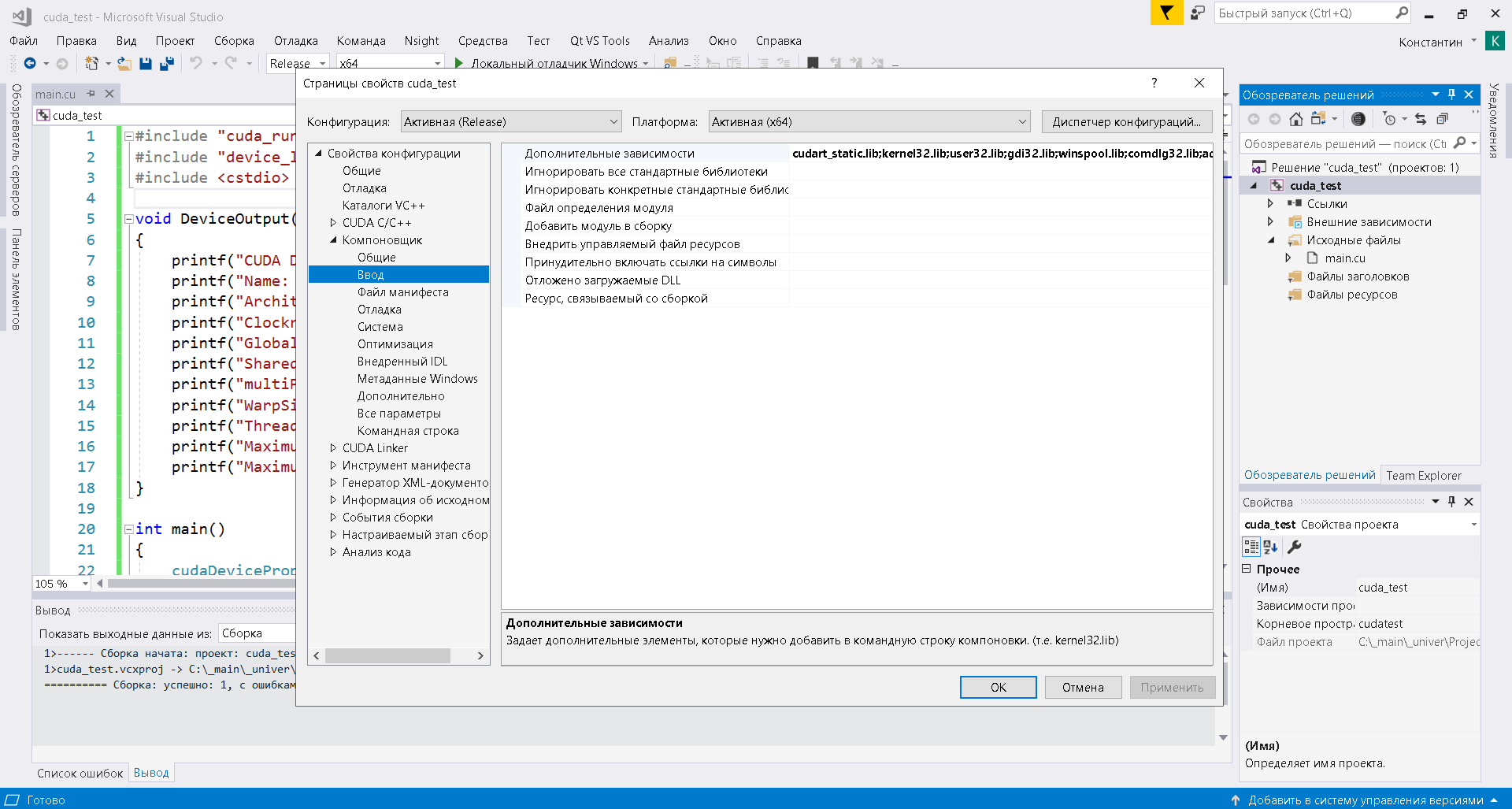


5.Меняем платформу на x64, копируем код из getDevices.txt, настраиваем пути до заголовочников и объектников:  
Стандартный путь до заголовочников:  
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\include  
Стандартный путь до объектников:  
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\lib\x64

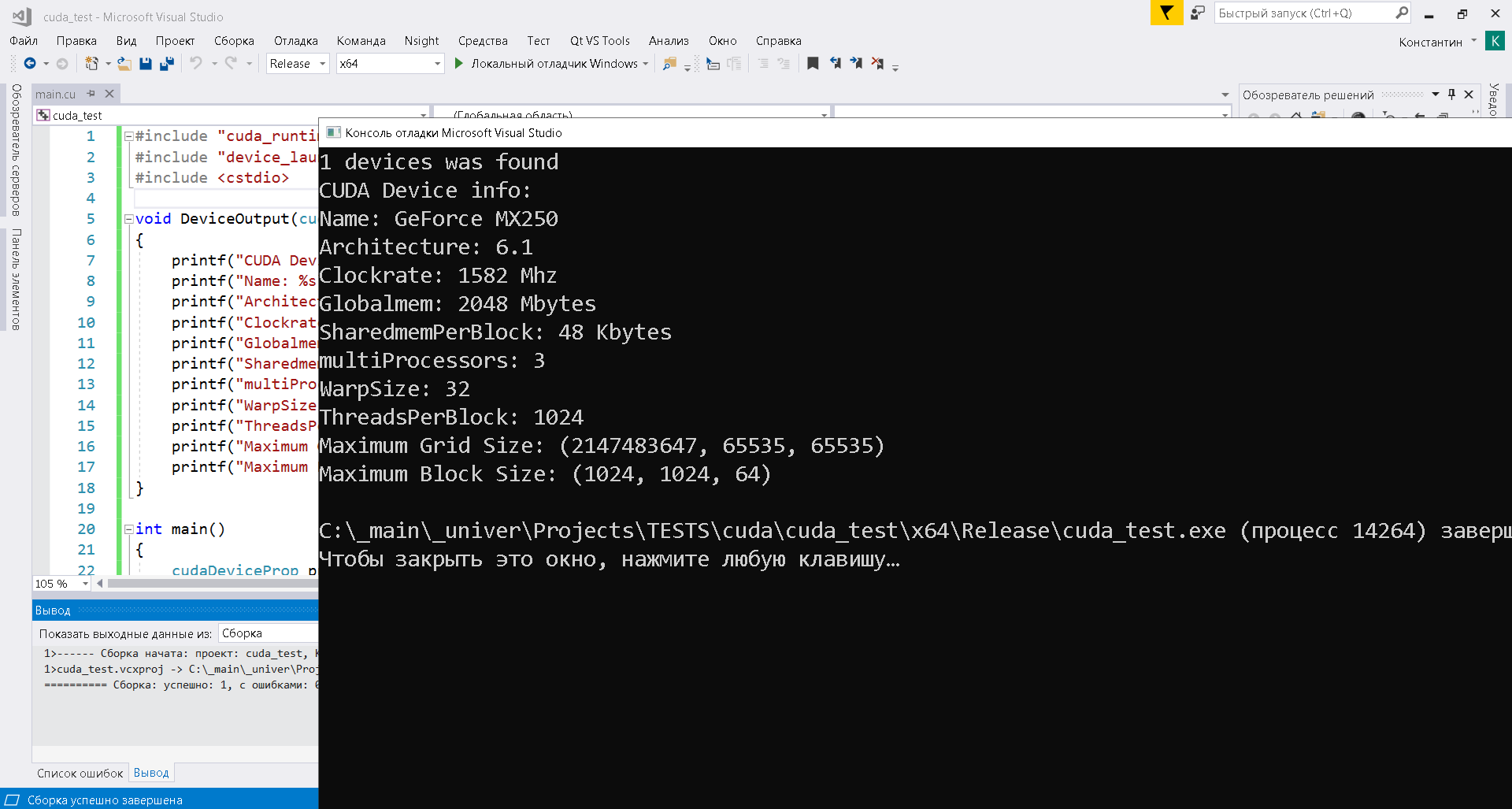
Соответственно путь до include пишем в Свойства проекта -> Каталоги VC++ -> Включаемые каталоги, а путь до объектников пишем в Свойства проекта -> Компоновщик -> Общие -> Дополнительные каталоги библиотек (иногда это уже сделано за вас благодаря шагу № 2).



Необходимо также прописать имя объектника cudart\_static.lib (если хотите статическую линковку) в Свойства проекта -> Компоновщик -> Ввод -> Дополнительные зависимости.



6.Билдим, запускаем, получаем информацию по устройствам CUDA.



Потенциальные проблемы: иногда выставленные по умолчанию настройки Свойства проекта -> CUDA C/C++ -> Device -> Code Generation compute\_30,sm\_30 не подходят и их нужно сменить на актуальные для вас (находим здесь <https://developer.nvidia.com/cuda-gpus> или здесь <https://en.wikipedia.org/wiki/CUDA> свою видяху и смотрим, какая у неё compute capability version. Например у MX250 версия 6.1, поэтому можно написать:

