**CUDA_examples REPOSITORY TUTORIAL:**

CUDA programs step by step tutorial for executing repository programs. It consist of tip dashes.

1) obtain computer with included NVidia GPU graphics ( programs are written in Nvidia corporation C extensions language with some C++11 capabilities - Compute Unified Device Architecture CUDA ). If you have not any, I will provide you some new-parts proposition in 2017y:

- CPU: AMD A10 7860k,
- MB: Asrock FM2A88X+ ( note PCIe 3.0 x16 bandwidth ~14GBps ),
- RAM: 4x GOODRAM PLAY Blue 4GB,
- pendrive 16GB with Ubuntu desktop 64bit, power supply,
- GPU: Nvidia GTX 1060 6GB
  - tip: consider operating system distribution booting to RAM ( f.e. modified Ubuntu ),
  - tip: consider good quality monitors, chair and keyboard ( for my best knowledge mouse is unnecessary ),
  - tip: for parallel cluster consider adding SSD RAID 0 and make redundand above configuration,
  - tip: theoretically low – price used parts R710 32GB + **gtx780** + PCIe: 2p 10GbE/IB/ 4p GbE aggregation – I did not check that configuration by myself, but should work,
  - tip: theoretical lower price – any used Personal Computer with PCIe 2.0x16 ( f.e. Core2Quad + 4GB RAM + **gtx580** ). I did not check that configuration by myself, but should work,

2) further consideration are for LINUX UBUNTU ( checked for 16.04 ). *I have been forced to checked that on Windows 10: Visual Studio Community ( 2015 ) and alternatively on Notepad++ with command line ( shell ), and both were working great with different commands.*

3) after signing in to operating system and making essential configurations ( f.e. second sudo user for recovery ) download CUDA toolkit from:

https://developer.nvidia.com/cuda-toolkit

3)to avoid:

- problems with lightdm log in ( login loop )
- problems with driver istall ( "Driver Installation failed: it appears, that a X server is running..." )

4) and succesfully install a NVidia CUDA Toolkit on Ubuntu 16.04 64bit I've just had to do:

- login on live session on pendrive ( "Try ubuntu, before install" )
- add sudo user at live session:
  #sudo adduser admin ( #pass: admin1 )
  #sudo usermod -aG sudo admin
- logout from live session, log in as #admin
- download CUDA Toolkit from NVidia official site ( ~1.5GB )
- change privileges for downloaded installer file ( DO NOT INSTALL AT THIS STEP! ):
  #sudo chmod +x cuda_X.X.run
- switch to console view:
  #Ctr+Alt+F1 ( to switch on terminal view )
  #Ctr+Alt+F7 ( to switch from terminal view to graphical server )
- at console view ( Ctr+Alt+F1 ) log in:
  #login: admin
  #pass: admin1

- stop graphical running service:
    #sudo service lightdm stop
- check if graphical server is off - after switching Ctr+Alt+F7 the monitor should be blank black, switch back on console view Ctr+Alt+F1
- install CUDA Toolkit, with such configuration:
    #sudo ./cuda_X.X.run
    #( press 'q' for license read skip )
    #do not install OpenGL library ( I do not know why – please do not ask )
    #do not update system X configuration
    #other options make yes and paths as default
- turn on graphical server:
    #sudo service lightdm start
- log in as user ( if you automatically log in as #ubuntu at live session log out ):
    #login: admin
    #pass: admin1
- check if nvcc exists:
    # sudo find /usr/ -name 'nvcc'

5) obtain git:
    #sudo apt-get install git
6) clone my repo in home directory ( ~ ):
    #git clone https://github.com/PiotrLenarczykAnonim/CUDA_examples.git
7) check whatever nvcc compiler works:
    #cd CUDA_examples/01_makeSimple/
8) most of folders are configured for #make via BASH scripts:
    #./RUN_COMMANDS.sh
9) Thrust library is delivered with CUDA Toolkit, but I strongly recommend for cloning repo Mr Jared Hoberock with examples:
    # git clone https://github.com/thrust/thrust.git
10) there are another my repo with C++ several examples:
    #git clone https://github.com/PiotrLenarczykAnonim/C-_examples.git

Post Scriptum: Mostly I have been developing it on Dell Inspiron 7746 with i7 5500U, 16GB RAM 1666MHz, Nvidia GM108M GeForce 845M and SSD 1TB.
Post Post Scriptum: Feel free to use that tutorial “as is“ without copyrights ( author Piotr Lenarczyk ) and of course guarantee.