

# Zadání semestrálního projektu

Jiří Filipovič

podzim 2016

# Hodnocení studentů a testových otázek

Úkolem je implementovat kernel počítající průměrné hodnocení studenta a průměrnou úspěšnost testové otázky

- vstupem je pole  $R$  o rozměrech  $q \times s$ , kde  $q$  je počet otázek a  $s$  počet studentů (každý řádek tedy obsahuje informace o studentovi, každý sloupec o otázce)
- výstupem jsou pole  $S$  a  $Q$ , obsahující průměrnou úspěšnost studentů a průměrný dosažený úspěch u otázek
- $S_i = \frac{\sum_{j=1}^q R_{i,j}}{q}$
- $Q_i = \frac{\sum_{j=1}^s R_{j,i}}{s}$
- hodnocení otázek v  $R$  je v procentech (celé číslo, rozsah 0-100), průměrné hodnocení ( $S$  a  $Q$ ) je počítáno s pohyblivou řádovou čárkou
- jednoduchý algoritmus, zajímavé optimalizace

# Implementace

Dostanete k dispozici "framework", do něhož stačí doimplementovat GPU výpočet

- automaticky nahraje vstupní data a uloží výsledky
- zkontroluje výsledky oproti CPU implementaci
- benchmarkuje (naivní) CPU a váš GPU kód
- lze upravovat pouze výpočetní funkci (soubor `kernel.cu`)
- build pomocí `nvcc -o framework framework.cu` (není třeba tvořit Makefile)

GPU funkce

- bude mít k dispozici pole vstupních a výstupních dat v GPU paměti
- vstupní a výstupní data nebudou zabírat více než polovinu GPU paměti (lze tedy alokovat struktury pro mezivýsledky)

# Pravidla projektu

Jak budeme měřit?

- velikost  $q$  a  $s$  vždy dělitelná 32
- výkon budeme měřit na computing capability 3.0 či vyšší, pro starší GPU nemusí být implementace funkční
- GPU a CPU výpočet by se neměl lišit více než o 0.000001 %

Co zakazujeme?

- spolupráci (obecné problémy lze diskutovat, preferujte prosím fórum v IS)

# Etapy projektu

Projekt je dělen do několika etap:

- první funkční implementace (25b): deadline 3.11.
- rozumně rychlá implementace (25b): deadline 29.11.
- závěrečná soutěž (až 20b podle umístění + dárek): deadline 12.12.

# První úkol

Napište první funkční paralelní implementaci.

- zatím nezáleží na rychlosti
- stačí funkčnost pro c.c. 3.0 a vyšší
- termín 3.11. (kdykoliv do konce dne)
- bodování dle funkčnosti odevzdaného řešení, za pozdní odevzdání -2 body za každých 24 hodin
- zkuste i optimalizovat (ať nejste pod přílišným časovým tlakem později)

# Další etapy

## 2. etapa

- rychlost alespoň 5 000 mevals/s při  $s = q = 2048$  na airacuda

## 3. etapa

- dosažení nejvyšší možné rychlosti, budou testovány i vstupy různé velikosti (i s  $q \ll s$  či naopak)