

Subiecte PGPU – 11 Febr 2k21

16 întrebări – 60 minute

Question 1 Not yet answered Marked out of 0.50 Flag question	<p>În CUDA, accesul unui thread la memoria shared este mai rapid decât la memoria globală deoarece</p> <div></div>
-----------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------

Question 2 Not yet answered Marked out of 0.50 Flag question	<p>În CUDA, modificatorul <input type="text"/> marchează o funcție kernel.</p> <div><input type="text"/>__kernel__ <input type="text"/>__device__ <input type="text"/>__unified__ <input type="text"/>__global__ <input type="text"/>__shared__ <input type="text"/>__local__</div>
-----------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 3 Not yet answered Marked out of 0.50 Flag question	<p>În CUDA, id-ul unui thread poate fi accesat prin variabila internă</p> <p>Answer: <input type="text"/></p>
-----------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------

Question 4 Not yet answered Marked out of 1.00 Flag question	<p>Fie următoarea secvenă de cod:</p> <pre>__global__ void my_kernel(int *out) { __shared__ int my_shared_variable; if(threadIdx.x == 0) my_shared_variable = 0; __syncthreads(); my_shared_variable += threadIdx.x; __syncthreads(); if(threadIdx.x == 0) out[blockIdx.x] = my_shared_variable; }</pre> <p>Care vor fi valorile vectorului out după terminarea execuției acestui kernel, considerând că s-a lansat în execuție un grid de 4 blocuri cu 256 de thread-uri fiecare (gridul și blocul sunt organizate liniar). Justificați.</p> <div></div>
-----------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 5 Not yet answered Marked out of 0.50 Flag question	<p>Ce este un branch divergent și cum este tratat pe GPU?</p> <div></div>
-----------------------------------------------------------------------	---------------------------------------------------------------------------

Question 6 Not yet answered Marked out of 0.50 Flag question	<p>Un multiprocesor de pe un GPU suportă 8K regiștri și până la 768 de thread-uri. Câte warp-uri pot fi executate concurrent dacă se utilizează blocuri de 512 thread-uri și fiecare thread necesită 16 regiștri? Justificați.</p> <div></div>
-----------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 7 Not yet answered Marked out of 0.50 Flag question	<p>Care sunt diferențele între un thread GPU și unul CPU?</p> <div></div>
-----------------------------------------------------------------------	---------------------------------------------------------------------------

Question 8 Not yet answered Marked out of 1.50 Flag question	<p>Fie următorul kernel CUDA:</p> <pre>__global__ void accessFloat3(float3 *d_in, float3 *d_out) { int index = blockIdx.x * blockDim.x + threadIdx.x; float3 a = d_in[index]; a.x += 2; a.y += 2; a.z += 2; d_out[index] = a; }</pre> <p>Care este problema cu acest kernel? Explicați cum ați rezolva această problemă.</p> <div></div>
-----------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 9 Not yet answered Marked out of 0.50 Flag question	<p>Datați două exemple de operații de reducere și menționați două metode prin care se poate realiza o astfel de operație în paralel pe GPU.</p> <div></div>
-----------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 10 Not yet answered Marked out of 0.50 Flag question	<p>Menționați două tipuri de întârzieri (overhead) ce pot apărea în programele CUDA.</p> <div></div>
------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------

Question 11 Not yet answered Marked out of 0.50 Flag question	<p>Care afirmație este adevărată?</p> <p><input type="radio"/> a. CUDA permite cooperarea între thread-uri</p> <p><input type="radio"/> b. un thread CUDA are acces R/W la memoria constantă</p> <p><input type="radio"/> c. CPU și GPU au un spațiu fizic de memorie comună</p> <p><input type="radio"/> d. un kernel CUDA este executat pe host</p>
------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 12 Not yet answered Marked out of 0.50 Flag question	<p>Memoria a unui GPU este dispusă on-chip și poate fi accesată mult mai rapid decât memoria DRAM</p> <p>Answer: <input type="text"/></p>
------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------

Question 13 Not yet answered Marked out of 1.00 Flag question	<p>1 punct din oficiu</p> <p>INTRODUCETI MINIM UN CARACTER pentru a primi punctul din oficiu.</p> <p>Answer: <input type="text"/></p>
------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------

Question 14 Not yet answered Marked out of 0.50 Flag question	<p>Care afirmație este adevărată?</p> <p><input type="radio"/> a. o variabilă de tip __shared__ este vizibilă tuturor thread-urilor dintr-un bloc</p> <p><input type="radio"/> b. thread-urile CUDA nu pot coopera prin intermediul memoriei comune</p> <p><input type="radio"/> c. cudaMemcpy() este o funcție asincronă, înainte de startarea copierii datelor fiind necesară sincronizarea thread-urilor</p> <p><input type="radio"/> d. tipul de retur al unui kernel CUDA nu poate fi void</p>
------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 15 Not yet answered Marked out of 0.50 Flag question	<p>Fie următoarea secvență de cod:</p> <pre>int main(int argc, char *argv[]) { ... kernel1<<<GRID_SIZE, BLOCK_SIZE>>>(devParamIn, devParamOut); kernel2<<<GRID_SIZE, BLOCK_SIZE>>>(devParamIn, devParamOut); retVal = cpuFunction(hostParamIn, hostParamOut); ... }</pre> <p>Marcați cu ADEVĂRAT sau FALS următoarele afirmații:</p> <p>1. Funcția <i>cpuFunction</i> este apelată numai după terminarea execuției <i>kernel2</i> <input type="text"/></p> <p>2. Funcția <i>cpuFunction</i> este apelată imediat după apelul <i>kernel2</i>, fără a aștepta terminarea execuției acestuia <input type="text"/></p> <p>3. Cele două funcții kernel se execută în paralel (simultan) pe GPU <input type="text"/></p> <div><input type="button" value="ADEVĂRAT"/> <input type="button" value="FALS"/></div>
------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Question 16 Not yet answered Marked out of 0.50 Flag question	<p>Fie următoarele declarații/definiții:</p> <pre>__device__ int deviceVar; __global__ void someFunc(int *input, int *output) { int someVar; int someArray[10]; __shared__ int anotherVar[10]; }</pre> <p>În ce spațiu de memorie CUDA se alocă următoarele variabile?</p> <p>deviceVar: <input type="text"/></p> <p>someVar: <input type="text"/></p> <p>someArray: <input type="text"/></p> <p>anotherVar: <input type="text"/></p> <div><input type="button" value="global"/> <input type="button" value="shared"/> <input type="button" value="register"/> <input type="button" value="constant"/> <input type="button" value="texture"/></div>
------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------