GPUs and Heterogeneous Systems – A.Y. 2023-24

Scuola di Ingegneria Industriale e dell'Informazione
Prof. Antonio Miele

January 10, 2025 - **FIRST PART OF THE EXAM**

Surname:	Name:	Personal Code:

Question	1	2	3	4	5	6	OVERALL
Max score	3	3	3	3	3	3	18
Score							

Instructions:

- This first part of the exam is "closed book". The students are not allowed to consult any course material and notes.
- No extra devices (e.g., phones, iPad) are allowed. Please, shut down and store any electronic device.
- Students are not allowed to communicate with any other ones.
- Students can write in pen or pencil, any color, but avoid writing in red.
- Any violation of the above rules will lead to the invalidation of the test.
- **Duration: 40 minutes**

Question 1
Briefly explain what SGI RealityEngine is.

Question 2
Specify in which NVIDIA GPU architecture the following mechanisms were introduced for the first time:

- Dynamic parallelism: ___________
- Unified shader processor: ___________
- Multi Instance GPU (MIG) virtualization: ___________

Question 3
Simulate the following simple sum reduction kernel and show the `data` array's content at each loop iteration. Consider a grid with 2 blocks, each block with 4 threads; the initial content of the `data` array is:
`[0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3]`

```
#define STRIDE_FACTOR 2

__global__ void reduce(double* data) {
  int i = threadIdx.x * STRIDE_FACTOR;
  int base_i = blockDim.x * blockIdx.x * STRIDE_FACTOR;
  for (int stride = 1; stride <= blockDim.x; stride *= STRIDE_FACTOR) {
    if (threadIdx.x % stride == 0) {
      data[base_i + i] += data[base_i + i + stride];
    }
    __syncthreads();
  }
}
```

Question 4
Draw the roofline model for a computing system having the following characteristics:

- Peak computational throughput = 150 GFLOPS
- Peak memory bandwidth = 100 GB/second

Question 5

Briefly explain what this piece of OpenCL code will print on the screen.

```
/* For the exercise we assume no error may occur. */
/* Moreover, macro values are not relevant */

#include "CL/cl.h"
#include <stdio.h>
#define MAXPLATFORMS ...
#define MAXDEVICES ...
#define MAXSTRING ...

int main(){
  int i, j;
  char text[MAXSTRING];
  cl_platform_id platformIds[MAXPLATFORMS];
  cl_device_id deviceIds[MAXDEVICES];
  cl_uint numPlatforms, numDevices;

  clGetPlatformIDs(0, NULL, &numPlatforms);
  clGetPlatformIDs(numPlatforms, platformIds, NULL);
  for (i=0; i<numPlatforms; i++){
    clGetPlatformInfo(platformIds[i], CL_PLATFORM_NAME, MAXSTRING, text, NULL);
    clGetDeviceIDs(platformIds[i], CL_DEVICE_TYPE_GPU, 0, NULL, &numDevices);
    if(numDevices>0){
     printf("%s :\n", text);
     clGetDeviceIDs(platformIds[i], CL_DEVICE_TYPE_GPU, numDevices, deviceIds, NULL);
      for (j=0; j<numDevices; j++){
          clGetDeviceInfo(deviceIds[j], CL_DEVICE_NAME, MAXSTRING, text, NULL);
        printf("%s\n", text);
      }
    }
  }
  return 0;
}
```

Question 6

In the following snippet of code using OpenACC pragmas, how many times will `foo()` and `bar()` be executed? Motivate the answer.

```
#pragma acc parallel num_gangs(16)
{
  #pragma acc loop gang
  for (int i=0; i<n; i++) {
    bar(i);
  }
  foo();
}
```

GPUs and Heterogeneous Systems – A.Y. 2023-24

Scuola di Ingegneria Industriale e dell'Informazione
Prof. Antonio Miele

POLITECNICO
MILANO 1863

January 10, 2025 - **SECOND PART OF THE EXAM**

Surname:	Name:	Personal Code:

Question	OVERALL
Max score	13
Score	

Instructions:

- This second part of the exam is "open book". The students are allowed to use any material and notes.
- The students are allowed to use the laptop and the tablet. No extra devices (e.g., phones) are allowed. Please, shut down and store not allowed electronic devices.
- Students are not allowed to communicate with any other one or use Internet.
- Students can write in pen or pencil, any color, but avoid writing in red.
- Students can also use the laptop to code the test solution. In this case, please pay attention to the instructor's instructions to submit the test solution.
- Any violation of the above rules will lead to the invalidation of the test.
- **Duration: 1 hour**

The source code can be downloaded from the course page on WeBeep

Question 1

Accelerate all the functions invoked in the following program onto GPU **exploiting task parallelism as much as possible**. Additional instructions:

- Function `func3` has to be executed on the host (CPU) while all other ones on the GPU.
- It is required to write the CUDA code of the kernel function accelerating `func1`.
- For the sake of brevity, the source code of the other functions to be accelerated is omitted.
- Set the block size to 32 for all kernel calls.

```
#define DIM 1000

void func1(int* a, int* b, int* c, int par, int dim); /* input: a and b, output: c */
void func2(int* a, int* b, int* d, int dim); /* input: a and b, output: d */
void func3(int* c, int* e, int dim); /* input: c, output: e - TO BE EXECUTED ON THE HOST */
void func4(int* d, int* f, int dim); /* input: d, output: f */
void func5(int* c, int* f, int* g, int dim); /* input: c and f, output: g */

int main(int argc, char **argv) {
  int a[DIM], b[DIM], c[DIM], d[DIM], e[DIM], f[DIM], g[DIM];

  /* ... acquire a and b ... (code is omitted) */

  func1(a, b, c, 50, DIM);
  func2(a, b, d, DIM);
  func3(c, e, DIM);
  func4(d, f, DIM);
  func5(c, f, g, DIM);

  /* ... display e and g ... (code is omitted) */
  return 0;
}

void func1(int* a, int* b, int* c, int par, int dim) {
  int i;
  for(i=0; i<dim; i++)
    c[i] = a[i] * par + b[i];
}
```