Intensive Computation

Prof. A. Massini

June 4, 2021

End-of-term test

Student's Name

_____

***Matricola* number**

_____

Exercise 1 (5 points)	
Exercise 2 (4 points)	
Exercise 3 (4 points)	
Question 1 (4 points)	
Question 2 (4 points)	
Exercise 4 (4 points)	
Exercise 5 (4 points)	
Exercise 6 (4 points)	
Total (33 points)	

Exercise 1 (5 points) - GPU & CUDA

You need to write a kernel that operates on an ultrasound color image represented by a matrix of size **1440x1280x24** (color images: have 24 bits for representing the three colors red, green, blue with 8 bits). You would like to assign one thread to each matrix element. You would like your thread blocks to use the maximum number of threads per block possible on your device.

a) How would you select the dimensions of a **2D grid and 2D rectangular blocks** for your kernel, minimizing the number of idle threads? Consider a device having compute capability 1.3.

b) How would you select the dimensions of a **2D grid** and **3D blocks** with the three sides all equal for your kernel, minimizing the number of idle threads? Consider a device having compute capability 3.5.

Technical specifications	**Compute capability (version)**									
	1.0	**1.1**	**1.2**	**1.3**	**2.x**	**3.0**	**3.5**	**3.7**	**5.0**	**5.2**
Maximum dimensionality of grid of thread blocks	2				3					
Maximum x-dimension of a grid of thread blocks	65535					2^{31}-1				
Maximum y-, or z-dimension of a grid of thread blocks	65535									
Maximum dimensionality of thread block	3									
Maximum x- or y-dimension of a block	512				1024					
Maximum z-dimension of a block	64									
Maximum number of threads per block	512				1024					
Warp size	32									
Maximum number of resident blocks per multiprocessor	8					16			32	
Maximum number of resident warps per multiprocessor	24		32		48	64				
Maximum number of resident threads per multiprocessor	768		1024		1536	2048				
Technical specifications	**1.0**	**1.1**	**1.2**	**1.3**	**2.x**	**3.0**	**3.5**	**3.7**	**5.0**	**5.2**
	Compute capability (version)									

Exercise 2 (4 points) – Interconnection Networks – CLOS

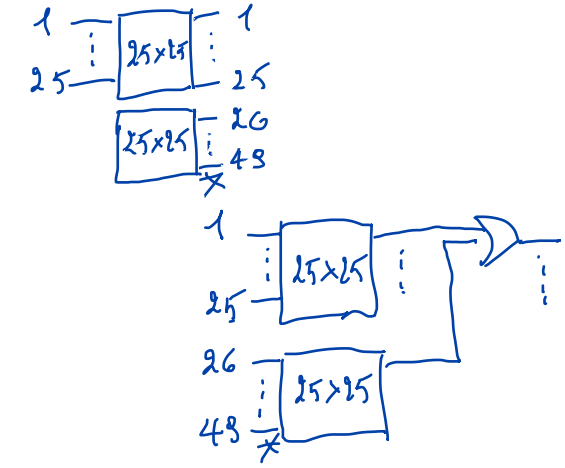
Design a Clos network of size 480 x 480, using in the first stage modules having 25 inputs. Consider both cases, strictly non-blocking and rearrangeable network.

Strictly non-blocking

$m \geq 2n - 1 = 2 \cdot 25 - 1 = 49$

$r = \left\lceil \frac{N}{n} \right\rceil = \left\lceil \frac{480}{25} \right\rceil = 20$

- stage 1 — 20 modules — 25×49
- stage 2 — 49 modules — 20×20
- stage 3 — 20 module — 49×25



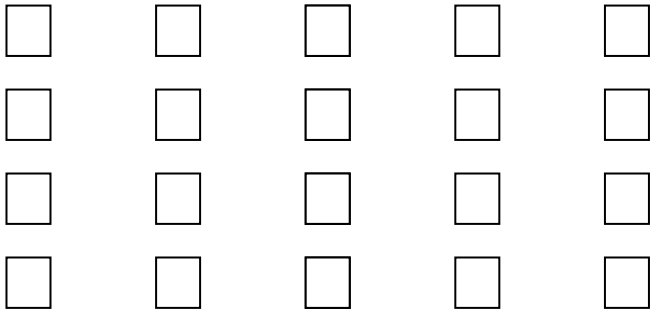
- stage

Compare the cost of the crossbar 480 x 480 and the Clos network, strictly non-blocking and rearrangeable non-blocking, designed in the previous point.

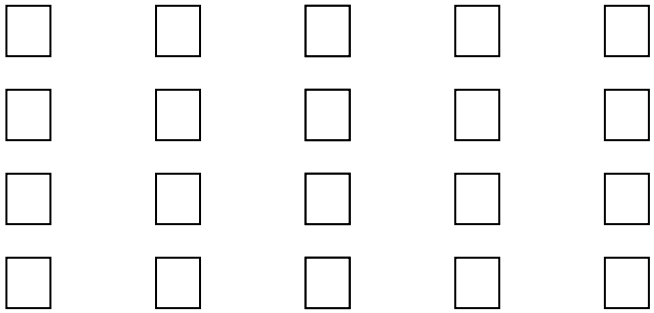
$\frac{35000}{} = 6,58$

Exercise 3 (4 points) – Interconnection networks – (2 log N - 1) MIN

Draw a Shuffle-Shuffle network of size N=8 using the switches below. Is it possible to realize the permutation $P = \begin{pmatrix} 01 & 23 & 45 & 67 \\ 46 & 32 & 75 & 01 \end{pmatrix}$ setting all the switches in the central stage on the straight state?



Which multistage network with 2logN -1 stages would you use to route the permutation P above, that guarantees to find a solution? Draw the network and show the connections.



Question 1 (4 points)

Briefly explain the **unit-gate model** giving the gate-count and gate-delay, used to evaluate time and area of circuits based on gates AND, OR, NOT and other gates derived from them.

Question 2 (4 points)

Explain what CUDA threads are and how they are organized in terms of blocks and grids. Describe how to obtain a unique ID for each thread by using the block ID and thread ID, in the case of a 2D grid and 3D blocks.

Exercises 4 (4 points) Amdhal Law

The following measurements are recorded with respect to the different instruction classes for the instruction set running a given set of benchmark programs:

Instruction Type	Instruction Count (millions)	Cycles per Instruction
Arithmetic and logic	5	6
Load and store	8	3
Branch	6	5
Others	6	4

Assume that instructions "*Branch*" can be modified so that they take 4 cycle per instruction instead of 5 as in the table. Compute the speedup obtained by introducing this enhancement using the **Amdhal law**.

How many cycles should "*Arithmetic and logic*" instructions consist of to reach at least the same speedup obtained modifying the number of cycles of "*Arithmetic and logic*" instructions as above?

Exercises 5 (4 points) Performance equation

Suppose we have made the following measurements, where we are considering Arithmetic and FP (Floating Point) instructions:
Frequency of Arithmetic operations = 20%
Average CPI of Arithmetic operations = 4.2
Average CPI of other instructions = 2.8

Frequency of FP operations = 30%
CPI of FP = 5.6

Assume that the two design alternatives are to decrease the CPI of FP to 3.5 or to decrease the average CPI of Arithmetic operations to 2.4.
Compare these two design alternatives using the processor **performance equation** and compute the speedup in both cases.

Exercise 6 (2+2 points) - Circuit time and area

A particular way to implement an adder is given by the ***carry-select adder***, generally consisting of ripple carry adders and multiplexers. A 8-bit carry-select adder with a uniform block size of 4 is shown in the figure.

i) Compute the ***time*** (propagation delay) and ***area*** required by the 8-bit carry-select adder.

ii) Compare the 8-bits carry-select adder and the standard binary ripple-carry adder (that is compute the **speedup** both for time and area).