



UNIVERSITY OF LIMERICK  
jou kennis vennyóú k knowlédgærtner

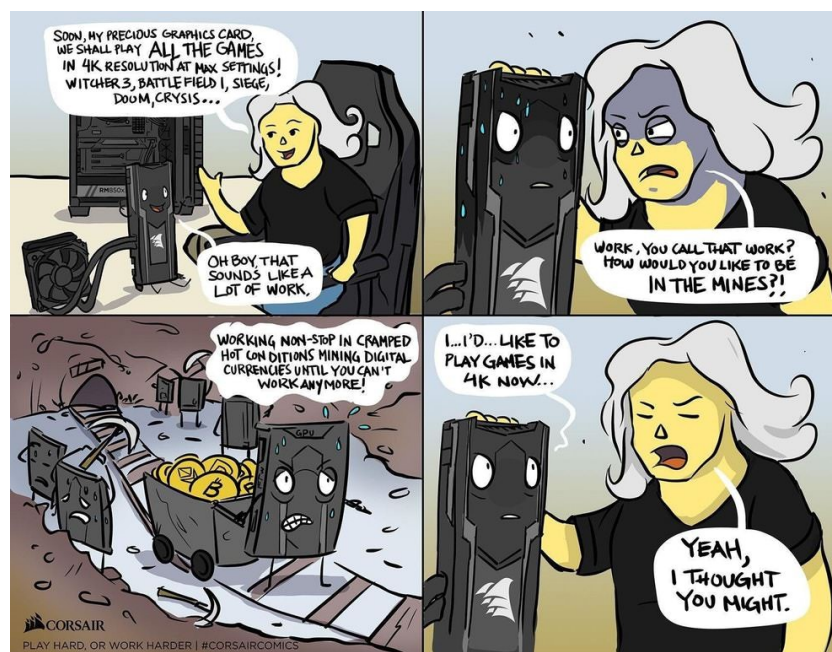
## Computer Systems 414

### Practical 8

2021

#### Aim of Practical 8:

1. To get practical practical experience of the use of hardware accelerators on desktop systems.



### **Rules of Engagement**

1. Attendance is required for this practical – if you cannot attend the practical in person you will have to find the OpenCL driver for your home computer and install them.
2. Students are allowed to work in groups of two, but each student must write and submit their own code.
3. Supporting documentation is available on learn.sun.ac.za, but students also have to do their own research.
4. The report is to be submitted on learn.sun.ac.za. The deadline for the reports is (14:00) Wednesday, 08 June 2021. **No late submissions will be accepted.**
5. Do not forget to cite and give credit for any information reported which is not your property.
8. Google is your friend. Any information not given is left out on purpose. Search for your solution and on the internet or relevant manuals and documentation..

## Assignment 8A

The practical of today will utilize Python to run a median filter both on the PC and on your mobile device. In the case of the PC we will use OpenCL (Google it!) to use the graphics processor (GPU) on the PC as a hardware accelerator to speed up the processing.

### Task 1 on the PC (Use the uploaded files from SunLearn as required)

1. Download Python 2.7 and install.
2. Download and install the Intel OpenCL SDK. Choose recommended settings everywhere and reboot the PC when done.
3. Navigate to c:\Python2.7\Scripts and open a terminal there (Total Commander rocks at this). Use pip to install *SciPy*, *Pillow* and *PyOpenCL*
4. Download the Python median filter example and the test image (you are welcome to use any image of your choosing, as long as it is large enough). Modify the code to add timing measurements to the median filter loop. Read the comments in the code to find the place and use something such as the `time.time()` routine from the Python *time* module.
5. Now repeat the measurements using the Numpy median filter module - Google for sample code (you can also use the OpenCV median filter if you install OpenCV first).
6. Include your measurements in your report and explain what you see.
7. What is 'General purpose computing on GPU' and what is the role of OpenCL in this?
8. Investigate the OpenCL median filter kernel code in the python file. What is the size of the median filter kernel?
9. Research the different types of median filter algorithms. How efficient is the algorithms that the OpenCL and Numpy code uses?

### Assignment 8B (Optional - for those who need more experience with VHDL)

1. Consider the following state diagram.

Present state $y_2 y_1$	Next state		Output $z$
	$w = 0$	$w = 1$	
	$y_2 y_1$	$y_2 y_1$	
0 0	1 0	1 1	0
0 1	0 1	0 0	0
1 0	1 1	0 0	0
1 1	1 0	0 1	1

2. Draw a flow-diagram for the above.
3. Write the VHDL code to implement the state diagram.
4. What will a testbench look like that will test your code?