High-performance computing with Python 1

Lab Exercise

Download the python file (ex1\_serial.py) from ftp://130.39.13.164/ex1\_serial.py. Save it to a local folder in your computer. Download the zip package from ftp://130.39.13.164/dem.zip and unzip it to the same folder. This is an ASCIIGRID Digital Elevation Model (DEM) of a mountainous area near Vancouver, Canada. You can open it in ArcMap.



Figure 1. The downloaded DEM data open in ArcMap.

Find *IDLE (Python GUI)* from the Start menu of Windows, and click to open it. Then, you will see a Python shell console like the below:



Figure 2. The Python console.



Figure 3. The Python program open in the editor.

Click *File* on the menu bar and then click *Open*. Browse to ex1\_serial.py and open it.

This program will perform a focal filter to a raster. For each pixel, the focal filter calculates the mean of the neighborhood pixels, and replaces the pixel’s value with the mean. This focal filter is like a smoother that smooth out the sharp variations in the raster. In this program, the neighborhood range is 3. The larger neighborhood range you use, the stronger the smoothing effect is.



Figure 4. the neighborhood of a pixel

**Task 1**: Click *Run* on the menu bar of the editor, and click *Run Module*. Alternatively, you can press F5 in your keyboard to run the program. During processing, you can press “Shift+Ctrl+Alt” keys together and then click “Start Task Manger” to launch Windows Task Manager. In the ‘Performance’ tab, you can observe the CPU usage, which is only around 25%, meaning that only one of the four virtual cores in your computer is running. You may need to wait for a few minutes for the program to run. When the program is done, a new DEM file (dem2.asc) will be generated. You can open this file in ArcGIS, where you can see the output DEM map has a smoother surface than the original one. The Python console will display the total processing time. Please write down your processing time here.

5. Download ex1\_par.py from ftp://130.39.13.164/ex1\_par.py and save it to the same folder of the DEM data. Open this file in the Python editor. This program does the same job as the previous one. However, it split the study area into a few sub-regions and run the mean filter simultaneously in different cores of your computer. In the program, the parameter *n* (the line below) specifies the number of cores used to run the program. Look at the program and try to understand the how this computing task is parallelized.



**Task 2**: Run the program and write down the processing time. Then, try to fill the following form by using different *n* value (i.e. different number of processors). During the program running, launch “Windows Task Manager” to observe the CPU usage.

|  |  |  |  |
| --- | --- | --- | --- |
| Number of cores | Processing time | Speedup | Efficiency |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 8 |  |  |  |