# Lab 1: Profiling Tools for Jetson Nano

**NOTE:** We will be installing essential software dependencies and packages so make sure you have a <u>stable internet connection</u> for your Jetson Nano before proceeding.

**Objective:** This lab will cover the installation and usage of essential profiling tools used for the Jetson Nano Development Kit.

#### Preface:

Profiling tools allow you to analyze and optimize the performance of your applications running on the Jetson Nano. This can be especially important for applications that require real-time processing or have strict latency requirements. Some of the benefits of using profiling tools on Jetson Nano devices include:

- Identifying performance bottlenecks
- Optimizing code for power consumption
- Finding and fixing bugs
- Testing and validating performance improvements

We will be visiting some of these ideas and getting familiar with the jetson-stats profiling tool with our lab activity.

#### **Materials Required:**

- Jetson Nano Development Kit (basic setup complete)
- ethernet cable / WIFI module
- keyboard and mouse
- display with cables / secondary computer

## **Assignment Submission Instructions**

You will need to turn in the following for <u>full credit (10 + 2 bonus points)</u> on today's lab:

- One screenshot of jetson-stats application (1 pts)
- filled out chart(2 pts) and all questions answered in Jupyter file (9 pts)

All the items above turned in as a .zip file or uploaded to the Github

You can login to canvas on your jetson nano and turn in the assignment.

For Github submissions, create a new branch "Lab 1" to your repo - follow Lab 0 doc for more details.

### Part 0 - Setup OS

Follow the instructions from Lab 0 and System Setup.

# Part I - Installation Steps

Step 1: Install relevant packages

- forum link to install pip3: https://www.odoo.com/forum/help-1/how-to-install-pip-in-python-3-on-ubuntu-18-04-167715
- open terminal using sidebar or search or Ctrl + Alt + T
- run the following commands (you will be in root)
- Be careful with pip and pip3, they are different

```
sudo su
apt-get install python3-pip
apt-get install python-pip
pip3 install -U jetson-stats
pip3 install memory_profiler
pip install memory_profiler
```

• optional: update current package list using

```
sudo apt update
```

After installation, try to run pip list to check

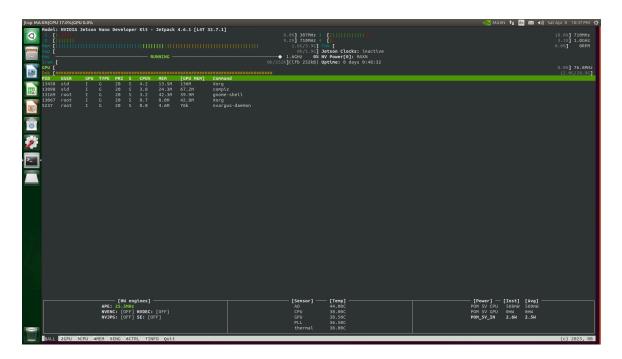
Step 2: Run jetson-stats

- repo and documentation for jetson-stats: <a href="https://github.com/rbonghi/jetson\_stats">https://github.com/rbonghi/jetson\_stats</a>
- simply run using itop

#### jtop

check to make sure program is functional

```
sktop:-5 sudo su
root@sid-desktop:/home/sid# apt-get install python3-plp
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
   apt-clone archdetect-deb bogl-bterm busybox-static cryptsetup-bin dpkg-repack gir1.2-timezonemap-1.0 g
kwin-data kwin-x11 libdeblan-installer4 libkdecorations2-5v5 libkdecorations2private5v5 libkf5activiti
libkf5declarative5 libkf5doctools5 libkf5globalaccel-data libkf5globalaccel5 libkf5globalaccelprivate5
   libkf5klocores libkf5klontlms libkf5klowidgetss libkf5newstuff-data libkf5newstuff5 libkf5newstuff5 libkf5newstuff5 libkf5sewstuff5 libkf5sewstuff5 libkf5textwidgets-data libkf5te libkf5semlguis libkscreenlocker5 libkwin4-effect-builtins1 libkwineffects11 libkwinglutils11 libkwinxre
   libqtSmultimediaS-plugins libqtSmultimediaquick-pS libqtSmultimediawidgetsS libqtSopenglS libqtSquickw
python3-dbus.mainloop.pyqt5 python3-icu python3-pam python3-pyqt5 python3-pyqt5.qtsvg python3-pyqt5.qt
    tasksel tasksel-data
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
python-pip-whl python3-setuptools python3-wheel
 Suggested packages:
   python-setuptools-doc
 Recommended packages:
   python3-dev
   he following NEW packages will be installed:
   python-pip-whi python3-pip python3-setuptools python3-wheel upgraded, 4 newly installed, 8 to remove and 8 not upgraded. sed to get 1,777 kB of archives.
Do you want to continue? [Y/n] y
  et:1 http://ports.ubuntu.com/ubuntu-ports blonic/universe arm64 python-pip-whl all 9.0.1-2 [1,379 kB]
```



## Part II - Install Jupyter lab

#### Step 1: Install relevant packages

- open terminal using sidebar or search or Ctrl + Alt + T
- run the following commands

```
# Install some dependencies first
sudo apt install nodejs npm
sudo pip3 install packaging
sudo pip3 install setuptools wheel
# if above command not working, use below one
sudo pip3 install -U pip setuptools wheel
sudo pip3 install jupyter jupyterlab
# using following command to check if it's installed successfully
jupyter lab --generate-config
```

After successful installation, run jupyter lab

- run the following commands
- The browser will popup

```
jupyter notebook
# use Ctrl + c to shut down
```

Choose your file to open

**NOTE**: There is an In [] in front of each cell. If the number in brackets turn into \*, it means it is **running** this cell. If it turns to numbers, that means it is finished.

### Part III - Further Exploring Constraints

In this section you run specific code snippets and document the memory usage, latency and performance of your Jetson Nano.

**NOTE:** Download the Python notebook file from the google drive (link in Step 1). Make sure you can access and run them using your Jetson Nano itself. Follow the notebook and answer the questions to successfully complete this part.

This will require a stable internet connection and enough storage space.

Step 1: Download the Lab1.ipynb notebook file from Canvas

Basic tutorial here

**Step 2:** Follow the instructions on the Python notebook

 Profiling and Timing Code | Python Data Science Handbook (jakevdp.github.io)

# Part IV - Fill the following chart (2 pts)

Simply go through the jetson-stats application (with no applications or tasks in the background) and fill in the chart below.

Jetson Nano Model	NVIDIA Jetson Nano Developer Kit
Jetpack version	Jetpack 4.6.1 [L4t 32.7.1]
memory storage	1.5 GB / 3.9 GB
CPU temperature	41 C
GPU temperature	41 C
GPU shared ram	12.1 GB / 58.5 GB
# of ARMv8 Processor cores	4
power usage	2.5 W

Additional questions in the Jupyter Notebook (7 pts + 2 bonus pts)