# Ten Simple Rules

## for getting started with command-line bioinformatics

Brandies PA, Hogg CJ (2021)
PLOS Computational Biology 17(2): e1008645

# TEN SIMPLE RULES

01

COMPUTING TERMINOLOGY

02

TOOL/PIPELINE SELECTION

03

ESTIMATING RESOURCES

04

SELECTING PLATFORMS

05

SOFTWARE INSTALLATION

06

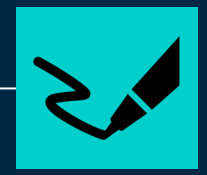
SCRIPT CURATION

07

MONITORING & OPTIMISATION

08

FILE MANIPULATION



09

RECORD KEEPING

10

PATIENCE!

01

# COMPUTING TERMINOLOGY

# GET FAMILIAR WITH COMPUTER TERMINOLOGY

ALGORITHM

DEPENDENCY

CPU

PIPELINE

HPC

VM

WALLTIME

EXECUTABLE

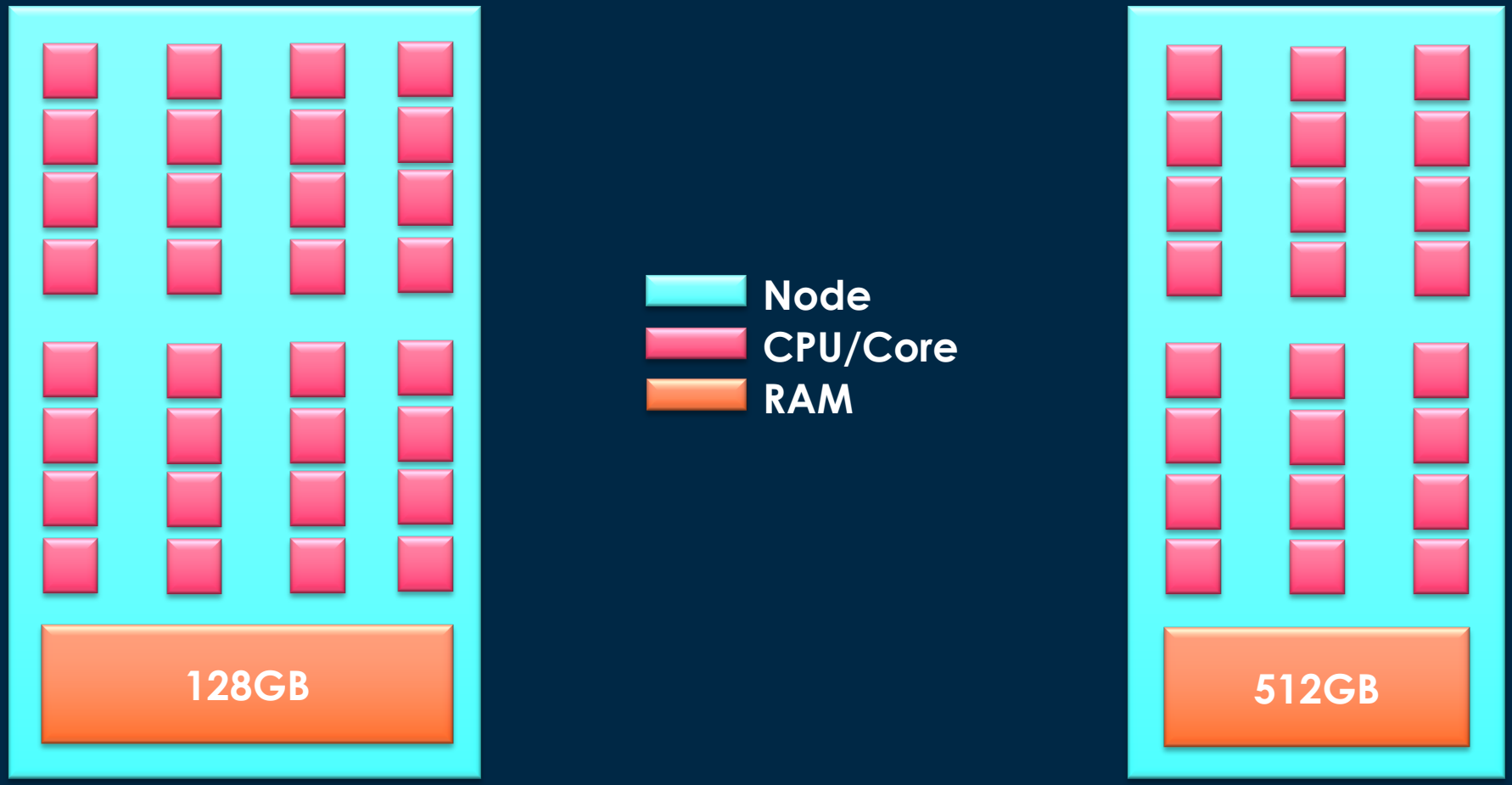
SCHEDULER

THREAD

RAM

NODE

# GET FAMILIAR WITH COMPUTER TERMINOLOGY




STANDARD

HIGH-MEMORY

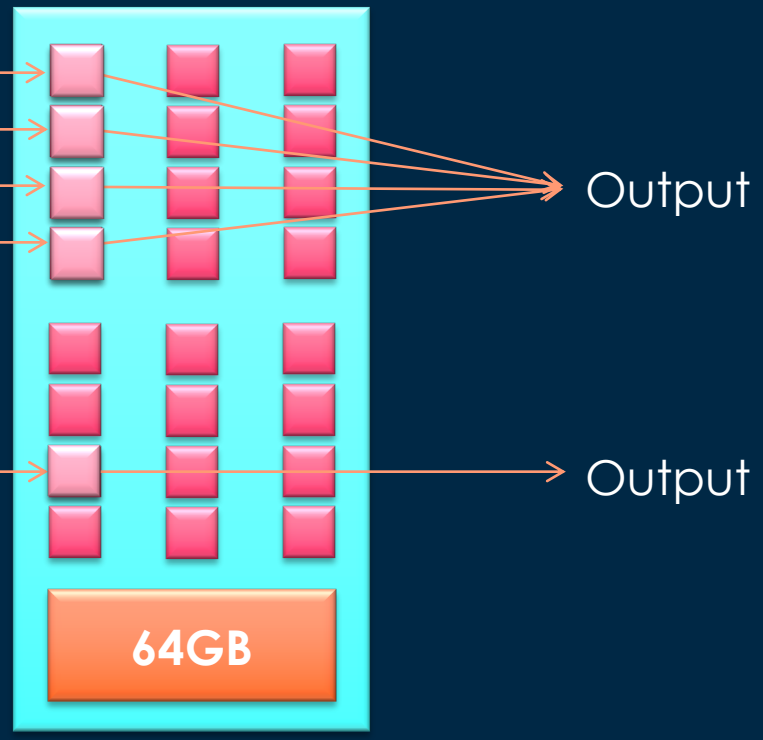
# GET FAMILIAR WITH COMPUTER TERMINOLOGY

**Multi-Threaded Program**:

- Process 1
- Process 2
- Process 3
- Process 4

**Single-Threaded Program**:

- Process 1 > Process 2 > Process 3 > Process 4



# 02

# TOOL/PIPELINE SELECTION

# KNOW YOUR DATA AND ASSESS YOUR NEEDS

- TARGET SPECIES/QUALITY OF DATA?
- AVAILABLE COMPUTING RESOURCES?
- AVAILABLE/TESTED TOOLS?

# KNOW YOUR DATA AND ASSESS YOUR NEEDS

https://australianbiocommons.github.io/2_tools.html

Home | Compute systems | Tools & workflows | Featured tools & workflows | Help! | Attributions

## Tool information

Column visibility

Search:

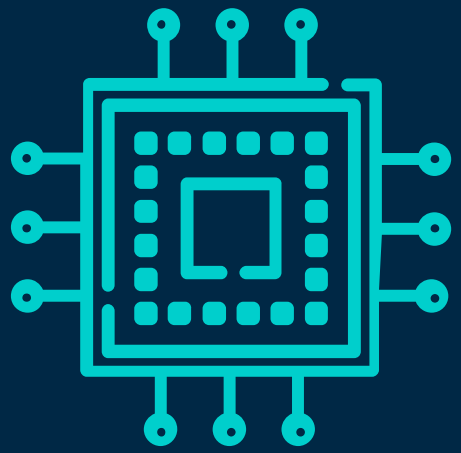
| | Tool / workflow name | Tool identifier (module name / bio.tools ID / placeholder) | Primary purpose (EDAM, if available) | Galaxy Australia | Available in Galaxy toolshed | NCI (Gadi) | Pawsey (Zeus) | Pawsey (Magnus) | QRIScloud / UQ-RCC (Flashlite, Awoonga, Tinaroo) | bio.tools | BioContainers |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 3D-DNA | 3D-DNA | Sequence assembly | | | | | | | | |
| 2 | ABRicate | abricate | Sequence analysis | Yes | abricate | | | | 1.0.1 | • | • |
| 3 | ABySS | abyss | Sequence assembly | | abyss | 2.2.3 | | | 2.0.2 | • | • |
| 4 | ALLMAPS | allmaps | Sequence assembly | | | | | | | | |
| 5 | allpathslg | allpathslg | Sequence assembly | | | | | | | | |
| 6 | amos | amos | Sequence assembly | | | | 3.1.0 | | | • | • |
| 7 | anaconda | anaconda | Gene structure | | | | | | 2019.07<br>2020.02<br>4.2.0<br>4.3.1<br>5.2.0 | • | |
| 8 | ANGSD | angsd | Population genomics | | | | | | 0.931<br>0.933 | • | • |
| 9 | ant | ant | | | | 1.10.7 | | | | | • |
| 10 | Apollo | apollo | Genomics / Genome annotation | | apollo | | | | | • | • |
| 11 | arima mapping | arima mapping | Chromosome conformation capture | | | | | | | | |

# 03

# ESTIMATING RESOURCES

# ESTIMATE YOUR COMPUTING REQUIREMENTS

- TOOL DOCUMENTATION + TEST DATASETS
- TOOL PUBLICATION + CITED PAPERS
- ASK COLLEAGES + USE ONLINE Q&A



32 CPUs
128GB RAM
1TB Storage

# ESTIMATE YOUR COMPUTING REQUIREMENTS

https://github.com/AustralianBioCommons

## IPA on Zeus @ Pawsey Supercomputing Centre

### Accessing workflow

The scripts for using IPA on Zeus have been made available below in the Quick start tutorial.

### Quick start tutorial

Below is a tutorial for install and annotation of IPA on Zeus.

**Note:** Purge_Dups is not included below as it requires a manual install. Documentation is currently not available for this, but may be in the future.

#### Install the improved phased assembler via bioconda

**Note:**

- the instructions below will install the latest version of miniconda and IPA.
- to install a specific version of IPA, use the following install script: `conda install pbipa=1.1.2`

**It is recommended to use an interactive session on zeus to install IPA. To launch an interactive session use:** `>salloc -n 1 -t 1:00:00`

1. Download miniconda: `curl -O https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh`
2. Install miniconda: `sh Miniconda3-latest-Linux-x86_64.sh`
   - **Note**: During the setup steps you will be asked where to install Miniconda. DO NOT install under `$HOME`, the suggested directory for miniconda is `/group/<project>/<user>/miniconda3`. Whenever you need to use miniconda from the command line you will need to run `export PATH=$PATH:/group/.../miniconda3/bin`, and in some cases also run `exec bash`

# SELECTING PLATFORMS

04

# EXPLORE DIFFERENT COMPUTING OPTIONS

| CLOUD | SHARED HPC |
| --- | --- |
| Customised computing resources | Fixed computing resources |
| Unshared resource (live programming) | Shared resource (scheduled jobs) |
| Complete control of compute environment | Compute environment largely controlled by IT |
| Both free and paid commercial options | Usually freely available based on institution/merit |

https://ronin.cloud

# 05

# SOFTWARE INSTALLATION

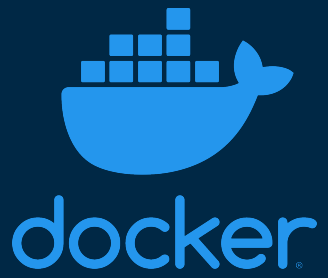
# UNDERSTAND THE BASICS OF SOFTWARE INSTALLATION

## PACKAGE MANAGERS

APT – DEBIAN

YUM – REDHAT

## CONTAINERS




## CONDA


## MANUAL INSTALL

Download + Unpack Code

↓

Configure Software

↓

Build Software

↓

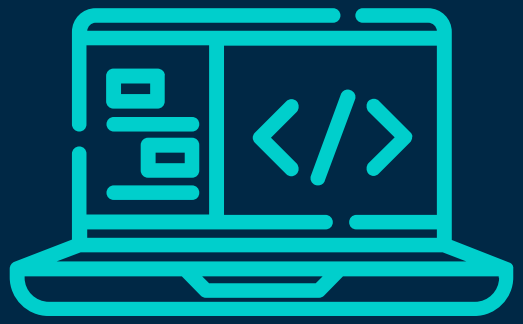
Install Software

# SCRIPT CURATION

06

# CAREFULLY CURATE AND TEST YOUR SCRIPTS

- SYNTAX OR MINOR SCRIPTING ERRORS
- SOFTWARE INSTALLATION ERRORS
- DEPENDENCY ISSUES
- SUBOPTIMAL COMPUTE RESOURCES



WHEN WORKNG WITH HPC ENVIRONMENTS, INTERACTIVE QUEUES ARE YOUR FRIEND!

# 07

# MONITORING & OPTIMISATION

# MONITOR AND OPTIMISE YOUR PIPELINES

## CLOUD VM

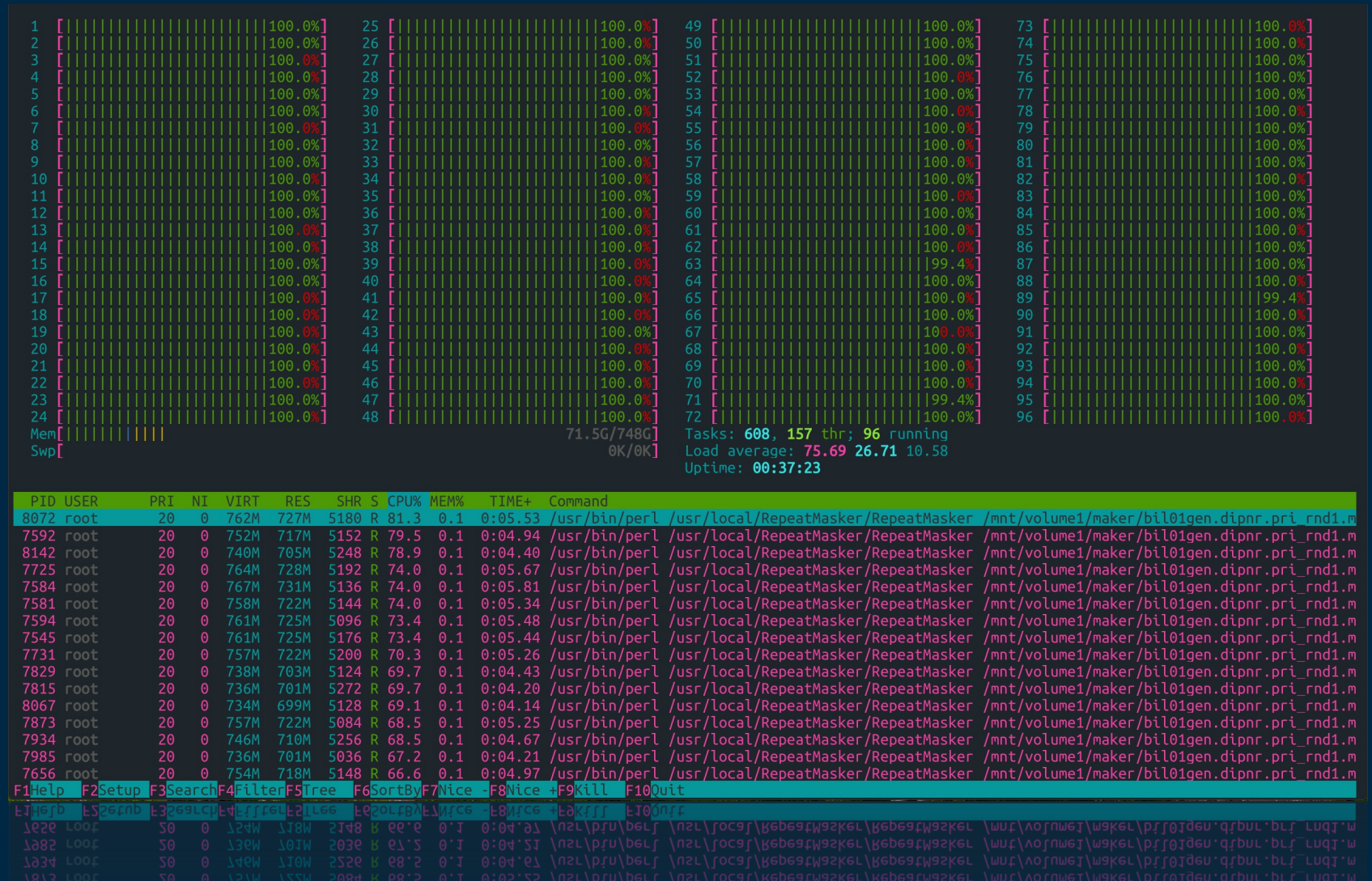
- **HTOP**
  Simple real-time monitoring of machine resources
- **NETDATA**
  Extensive monitoring using hundreds of preconfigured charts

## HPC

- **SCHEDULER LOGS**
  Summary of maximum and total compute resources used

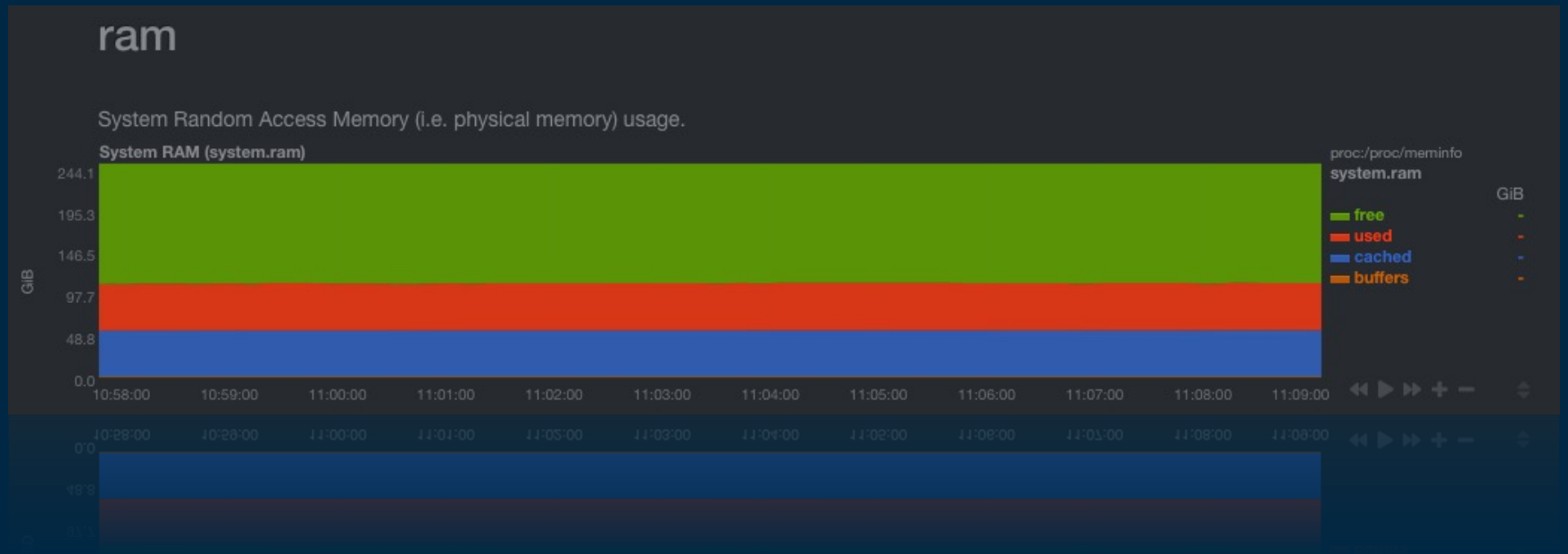
# MONITOR AND OPTIMISE YOUR PIPELINES

## HTOP



# MONITOR AND OPTIMISE YOUR PIPELINES

## NETDATA



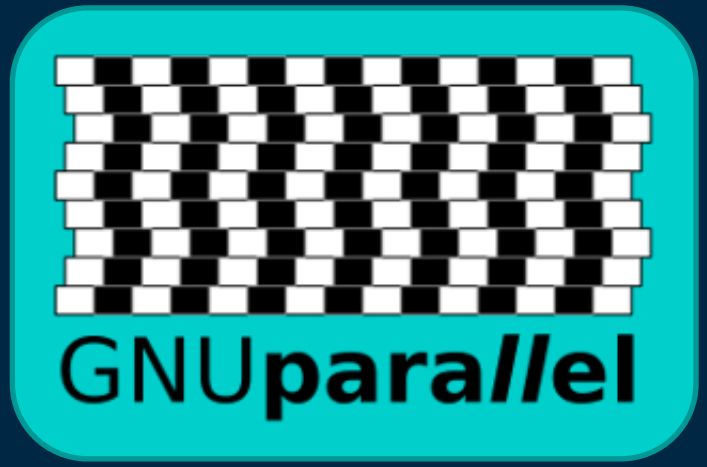
# MONITOR AND OPTIMISE YOUR PIPELINES

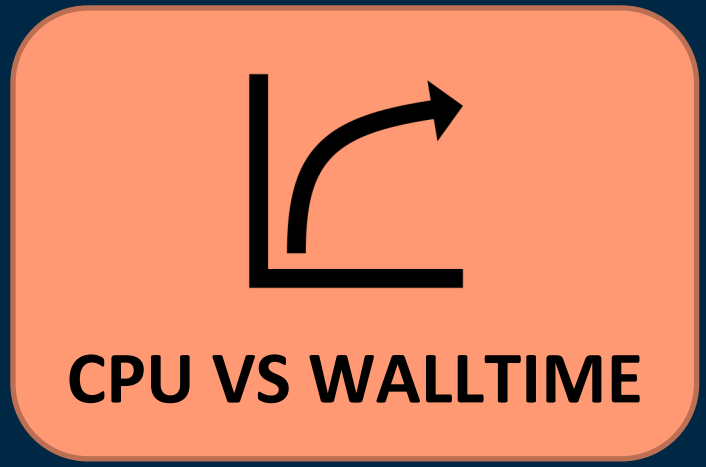
## SCHEDULER LOGS

```
Job Id: 2566026.pbsserver in queue normal
Job Name: variant_only_vcfs
Project: GENOMICS
Exit Status: 0
Job run as chunks (hpc140:ncpus=12:mem=100663296kb)
Walltime requested:  12:00:00 :
                                :
-- Nodes Summary --
-- node hpc140 summary
      Cpus requested:          12 :
            Cpu Time:    03:29:21 :
     Mem requested:       96.0GB :
                                  :
```

# MONITOR AND OPTIMISE YOUR PIPELINES

## OPTIMISATION



# 08

# FILE MANIPULATION

# GET FAMILIAR WITH BASIC BASH COMMANDS

## grep

Pattern Matching

```
grep "chromosome 5"
```

## awk

Data Processing

```
awk '$1 == 5 {print $2, $3}'
```

## sed

Find & Replace

```
sed 's/sample1/ID7037/g'
```

# RECORD KEEPING

09

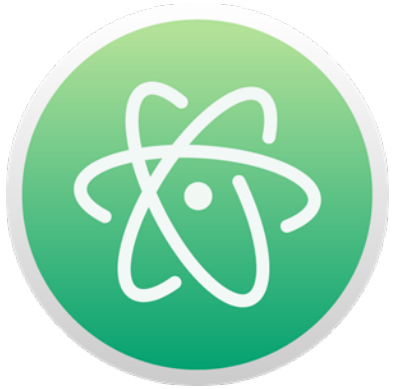
# WRITE IT DOWN!

labarchives Research Notebook

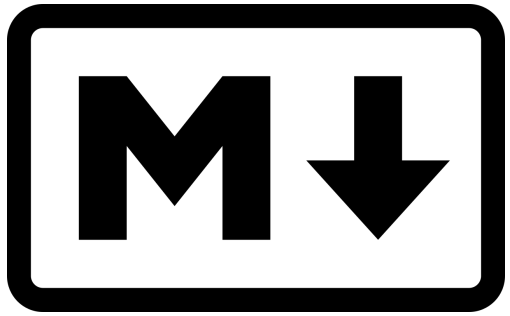
Evernote

git

VISUAL STUDIO CODE



ATOM



MARKDOWN

10

# PATIENCE

# PATIENCE IS KEY!

# THANK YOU! ☺


parice.brandies@sydney.edu.au

@PariceBrandies