## Summary of the tutorial

<table>
<thead>
<tr>
<th>Quick Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPC Home</td>
</tr>
<tr>
<td>Getting an account</td>
</tr>
<tr>
<td>Gentle Introduction to using HPC</td>
</tr>
<tr>
<td>Getting started on Prince</td>
</tr>
<tr>
<td>Prince How-to Articles</td>
</tr>
<tr>
<td>Logging in</td>
</tr>
<tr>
<td>Windows</td>
</tr>
<tr>
<td>Mac / Linux</td>
</tr>
<tr>
<td>Clusters and Storage</td>
</tr>
<tr>
<td>Prince (HPC)</td>
</tr>
<tr>
<td>Dumbo (Hadoop)</td>
</tr>
<tr>
<td>Brooklyn (OpenStack)</td>
</tr>
</tbody>
</table>
Dalmata (NYU Abu Dhabi)

Transferring data to/from the clusters

Transferring data to/from Prince cluster using Globus

Submitting jobs with sbatch

Available software

Licensed Software Available on the HPC Cluster

Building Software

Slurm Tutorial

Tutorials

FAQs

Scratch Area Cleanup

Programming for Biologists
Running jobs on the Prince Cluster

Accessing the Prince Cluster
  - From Windows workstation
  - From Mac workstation

Software and Environment Module

Job script and resource request
  - Introduction to job scheduling
  - Submitting jobs with sbatch
  - Requesting resources
• You can compile, edit scripts and view results on the login nodes, but **computational work should be run on the compute nodes**

• You can access compute nodes with `srun`
  
  • Either via a job script, or interactively
  
  • Compute nodes are allocated to jobs by the scheduler, so your job might not start immediately
  
  • Jobs must request resources, but mostly need not specify a queue.
  
  • Requesting just slightly more than when you expect to need is generally the best practice

• **Short jobs get higher priority, and short or small jobs are easier to schedule quickly**

• You can monitor your job’s progress with `squeue`, `sstat`, `sacct`, `scontrol` or `slurmtop`

• Software is managed by Environment Modules
  
  • Use `module avail` to find software packages
  
  • And `module load` to load them into your environment
    
    • including within job scripts!
  
  • Use `module purge` to return to a clean environment before loading a new set of modules
  
  • Other useful commands are `module list` and `module show`