

# Nexus Cluster

CMSC470



# CLIP Cluster

- CLIP Cluster is part of the broader cluster provided and maintained by UMIACS
- UMIACS manages SLURM to manage the cluster.

## Resources:

UMIACS Staff: [staff@umiacs.umd.edu](mailto:staff@umiacs.umd.edu)

UMIACS Wiki: [https://wiki.umiacs.umd.edu/umiacs/index.php/Main\\_Page](https://wiki.umiacs.umd.edu/umiacs/index.php/Main_Page)

SLURM Docs: <https://wiki.umiacs.umd.edu/umiacs/index.php/ClassAccounts#Overview>

# SLURM: What is it?

Simple Linux Utility for Resource Management

An open-source workload manager designed for Linux clusters of all sizes.

Job Queue



User Legend:  
■ User A  
■ User B  
■ User C  
■ User D

Compute Nodes



**Key Functions of SLURM:**

- 1. Resource Allocation**
  - Allocates access to computer nodes
  - Can be exclusive or non-exclusive access
  - Allocations are time-limited
- 2. Job Management Framework**
  - Starts, and executes, and monitors jobs
  - Typically handles parallel jobs
  - Handle dependent jobs.
- 3. Resource Contention Management**
  - Decides which jobs run first when the cluster is busy
  - Manages a queue of pending jobs

# How to Use Compute

Step 1:

Download Umiacs Virtual Private Network at

[https://itsupport.umd.edu/itsupport?id=kb\\_article\\_view&sysparm\\_article=KB0016076](https://itsupport.umd.edu/itsupport?id=kb_article_view&sysparm_article=KB0016076)

Install and Connect to GlobalProtect Virtual Private Network (VPN)

## Table of contents

- Install on macOS and Windows
- Connect on macOS and Windows
- Install and connect on iOS or Android
- Connect on Linux
  - Install instructions
- Connect on ChromeOS
- International student instructions
  - Add the access.umd.edu portal
  - Select the TunnelAll Gateway

# How to Use Compute

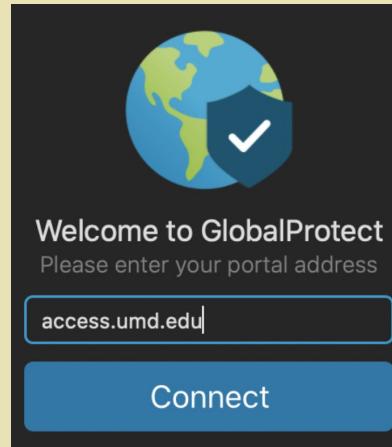
Step 1:

Download Umiacs Virtual Private Network at

[https://itsupport.umd.edu/itsupport?id=kb\\_article\\_view&sysparm\\_article=KB0016076](https://itsupport.umd.edu/itsupport?id=kb_article_view&sysparm_article=KB0016076)

Step 2:

Connect to VPN and authenticate



# How to Use Compute

Step 1:

Download Umiacs Virtual Private Network at

[https://itsupport.umd.edu/itsupport?id=kb\\_article\\_view&sysparm\\_article=KB0016076](https://itsupport.umd.edu/itsupport?id=kb_article_view&sysparm_article=KB0016076)

Step 2:

Connect to VPN and authenticate

Step 3:

Using your account- ssh login your account

Your assigned account:

{Your\_directory\_ID}@nexusclass00.umiacs.umd.edu

{Your\_directory\_ID}@nexusclass01.umiacs.umd.edu

# Accessing the Cluster: Login Nodes

SSH-ing to the Submission / Login Nodes:

>{Your\_directory\_ID}@nexusclass00.umiacs.umd.edu

>{Your\_directory\_ID}@nexusclass01.umiacs.umd.edu

Configure your login info in `~/.ssh/config`  
Follow the following steps

1. `nano ~/.ssh/config`
2. Edit nano file and add your account information to the file

```
Host nexusclass00.umiacs.umd.edu  
HostName nexusclass00.umiacs.umd.edu  
User {Your_Directory_ID}
```

# Accessing the Cluster: Login Nodes

SSH-ing to the Submission / Login Nodes:

>{Your\_directory\_ID}@nexusclass00.umiacs.umd.edu

>{Your\_directory\_ID}@nexusclass01.umiacs.umd.edu

Configure your login info in `~/.ssh/config`  
Follow the following steps

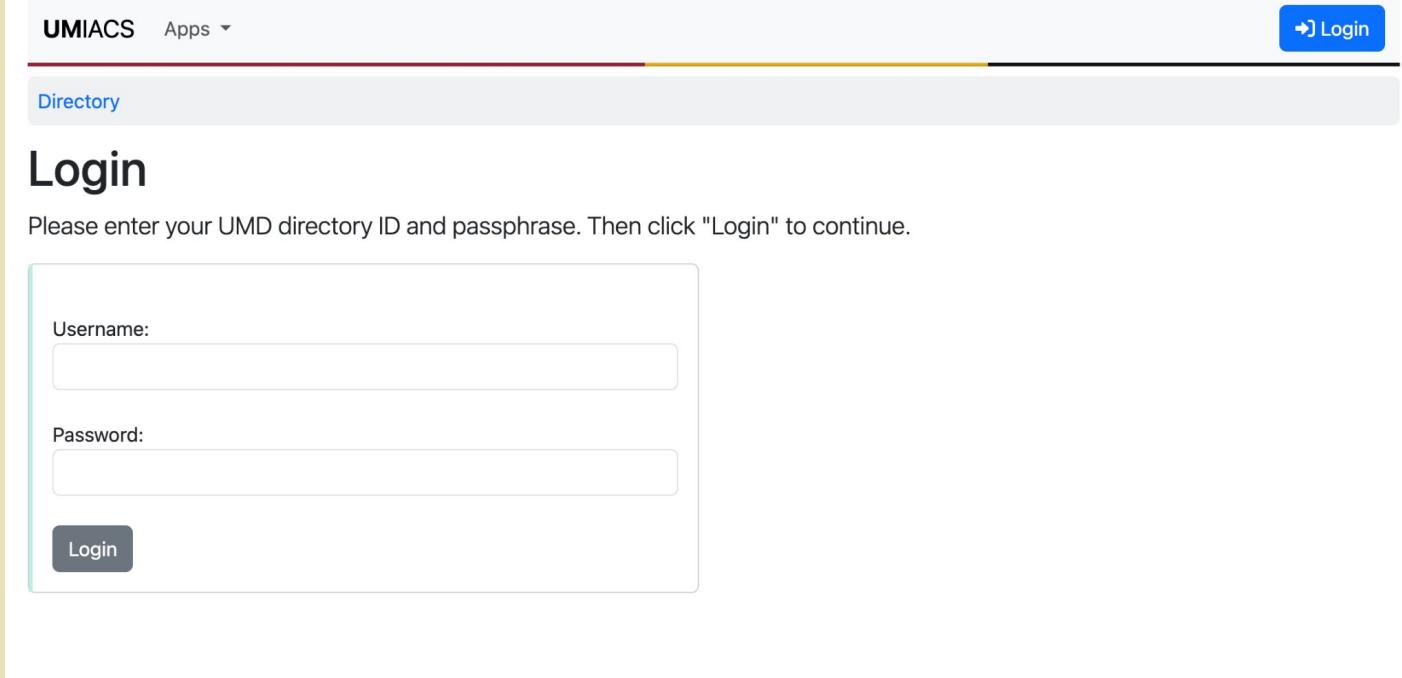
1. `nano ~/.ssh/config`
2. Edit nano file and add your account information to the file

```
Host nexusclass00.umiacs.umd.edu  
HostName nexusclass00.umiacs.umd.edu  
User {Your_Directory_ID}
```

3. Login through vscode or terminal

# Check your resource access

Login at <https://intranet.umiacs.umd.edu/directory/auth/login/?next=/directory/cr/>



The screenshot shows a web browser window for the UMIACS directory login. At the top, there is a header bar with the UMIACS logo, an 'Apps' dropdown, and a blue 'Login' button. Below the header, a navigation bar has 'Directory' selected. The main content area is titled 'Login' and contains instructions: 'Please enter your UMD directory ID and passphrase. Then click "Login" to continue.' It features two input fields: 'Username:' and 'Password:', each with a corresponding text input box. A large 'Login' button is positioned below the password field.

UMIACS Apps ▾

Directory

## Login

Please enter your UMD directory ID and passphrase. Then click "Login" to continue.

Username:

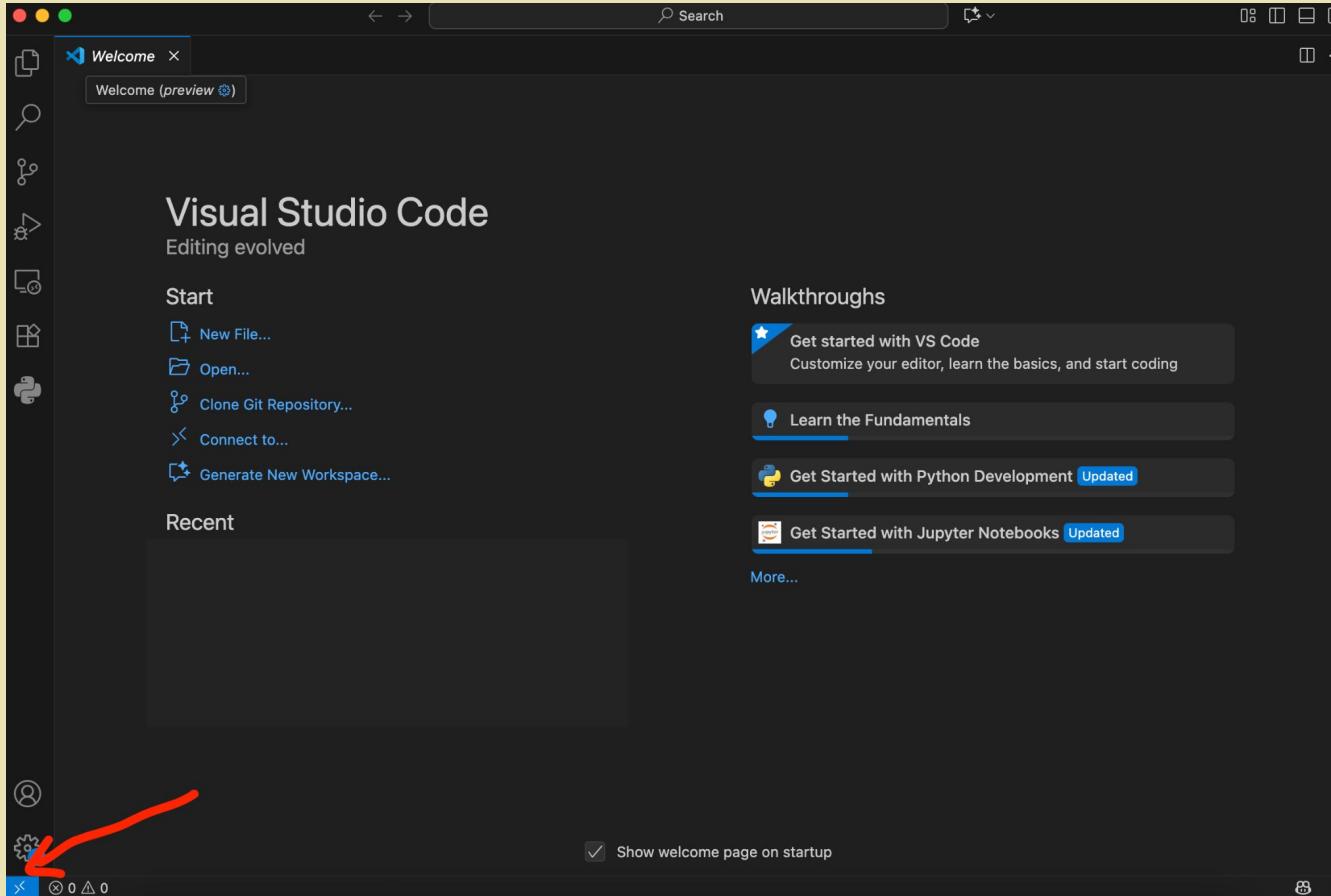
Password:

Login

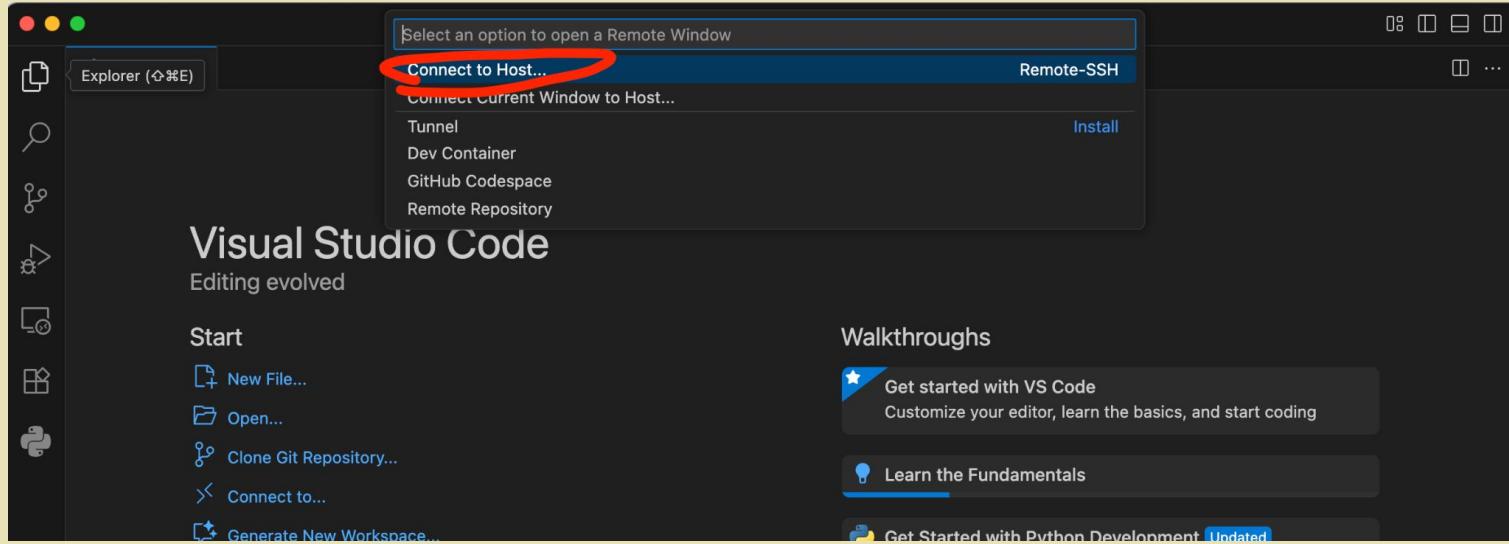
# Login Through Terminal

1. Open Terminal
2. ssh {Your\_Directory\_ID}@nexusclass00.umiacs.umd.edu
3. Enter your password
4. Access resource and code at terminal

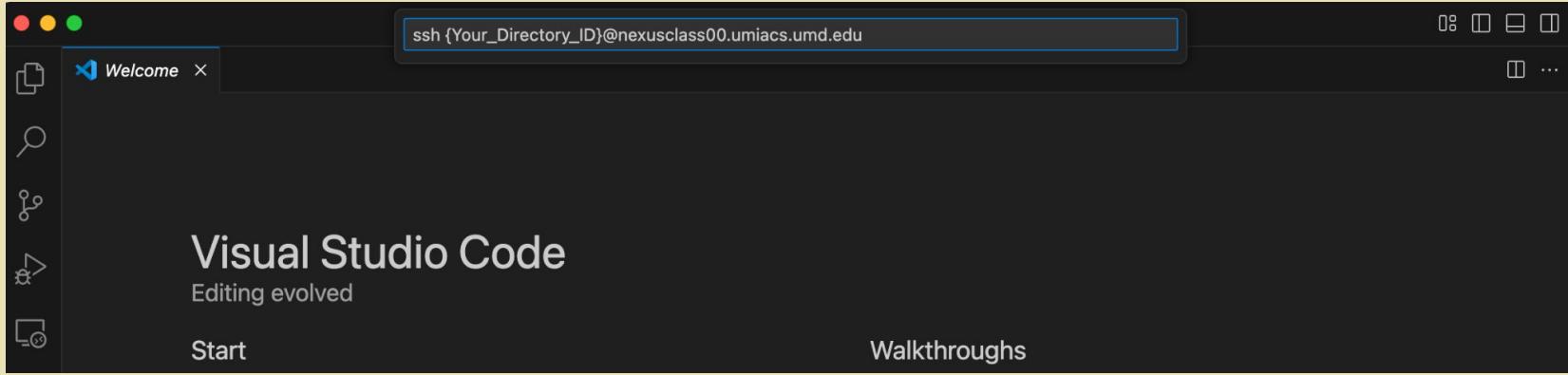
# Login through VS code



# Login through VS code



# Login through VS code



# After Loggin In

## 1. Install Conda and environment

<https://www.anaconda.com/docs/getting-started/miniconda/install#quickstart-install-instructions>

1. Run the following four commands to download and install the latest Linux installer for your chosen chip architecture. Line by line, these commands:
  - create a new directory named "miniconda3" in your home directory.
  - download the Linux Miniconda installation script for your chosen chip architecture and save the script as `miniconda.sh` in the miniconda3 directory.
  - run the `miniconda.sh` installation script in silent mode using bash.
  - remove the `miniconda.sh` installation script file after installation is complete.

```
64-bit AWS Graviton 2/ARM 64 IBM Z

mkdir -p ~/miniconda3
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh -O ~/miniconda3/miniconda3.sh
bash ~/miniconda3/miniconda3.sh -b -u -p ~/miniconda3/miniconda3
rm ~/miniconda3/miniconda3.sh
```

## 2. Activate conda

`source miniconda/bin/activate`

## 3. Create conda environment

`conda create -n environment_name python=3.11`

# Personal Storage

cd /fs/classhome/{Your\_Directory\_ID}

## Personal Storage

---

Your home directory has a quota of 30GB and is located at:

```
/fs/classhomes/<username>
```

You can request up to another 100GB of personal storage if you would like by **having your TA or instructor contact staff**. This storage will be located at

```
/fs/class-projects/<semester><year>/<coursecode>/<username>
```

where <semester> is either "spring", "summer", "fall", or "winter", <year> is the current year e.g., "2025", <coursecode> is the class' course code as listed in UMD's [Schedule of Classes](#) in all lowercase e.g., "cmsc999z", and <username> is your username.

# Requesting for GPU to run a bash script

On terminal, do

```
srun --qos=default --partition=class --account=class --time=2:00:00 --gres=gpu:1  
bash YOUR_SCRIPT.sh
```

Example bash script:

```
python run_eval.py
```

# Requesting Interactive GPU Session

```
srun --qos=default --partition=class --account=class --time=2:00:00 --gres=gpu:1
```

## Checking your GPU

nvidia-smi

```
(base) bash-4.4$ nvidia-smi
Tue Nov 11 23:23:43 2025
+
+-----+-----+-----+
| NVIDIA-SMI 580.95.05 | Driver Version: 580.95.05 | CUDA Version: 13.0 |
+-----+-----+-----+
| GPU  Name    Persistence-M | Bus-Id      Disp.A  | Volatile Uncorr. ECC |
| Fan  Temp    Perf          Pwr:Usage/Cap | Memory-Usage | GPU-Util  Compute M. |
|          %   %          /   /   |           /   /   |          %          MIG M. |
|-----+-----+-----+-----+-----+-----+-----+
|  0  NVIDIA RTX A4000     Off | 00000000:01:00.0 | Off       0%       Default |
| 41% 35C     P8          15W / 140W | 1MiB / 16376MiB |          0%          N/A |
|-----+-----+-----+-----+-----+-----+-----+
+
+-----+
| Processes:                               GPU Memory |
| GPU  GI  CI          PID  Type  Process name        Usage  |
| ID   ID          ID          |          |             |
+-----+
| No running processes found               |
+-----+
```