



**National Textile University**  
**Department of Computer Science**

**Subject**

Operating System

**Submitted to:**

Sir Nasir Mehmood

**Submitted by:**

Haider Ali

**Registration Number**

23-NTU-CS-1164

**Home Task**

01

**Semester**

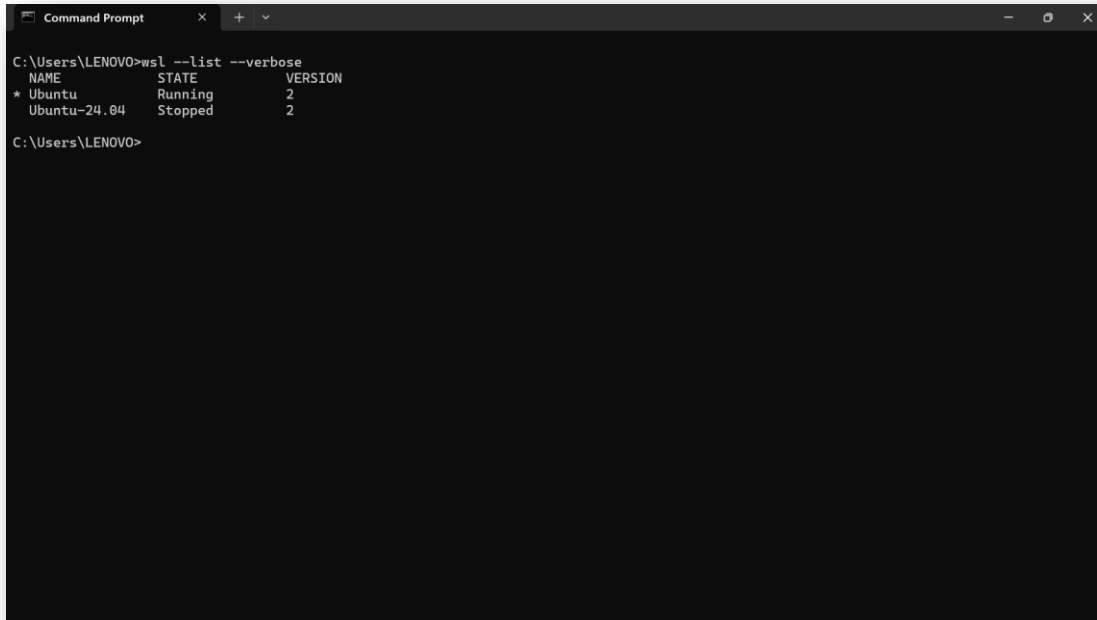
5th

## Part A: WSL2 & Ubuntu Setup

### 1. Verify WSL2 and Ubuntu installation

- Verify installation by running the following command in powershell:  
**wsl --list—verbose**

Screen Short:



```
C:\Users\LENOVO>wsl --list --verbose
  NAME      STATE      VERSION
* Ubuntu    Running     2
  Ubuntu-24.04 Stopped     2
C:\Users\LENOVO>
```

### 2. Updated Ubuntu environment

- Run the following command in Ubuntu:  
**sudo apt update && sudo apt upgrade -y**

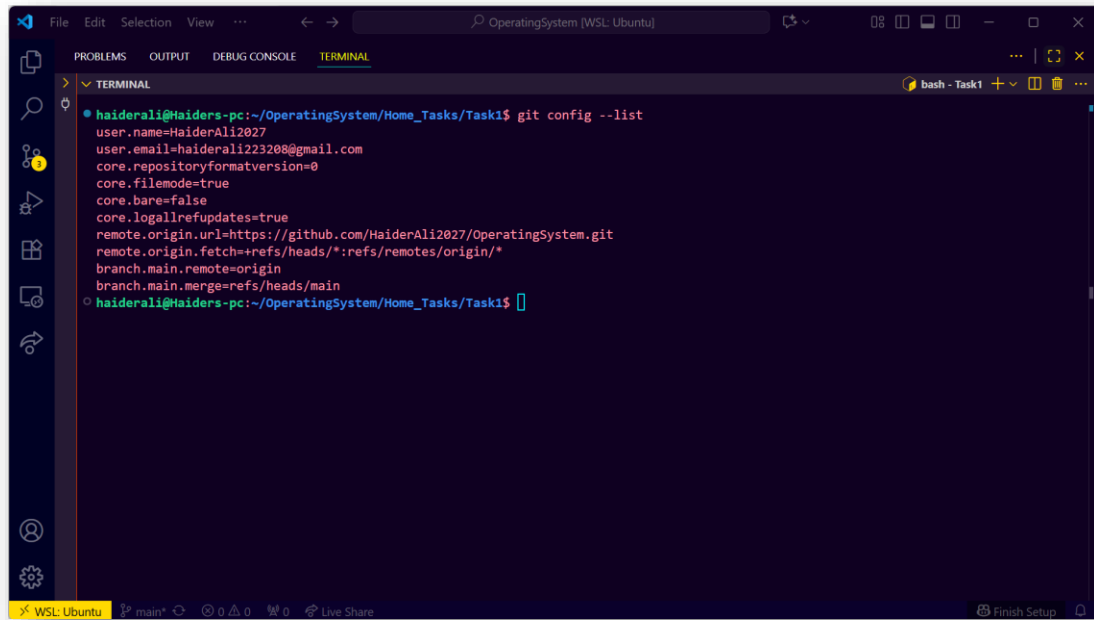
## Part B: Git & GitHub SSH Setup

### 1. Configure Git

- Set your name and email:  
**git config --global user.name "Your Name"**  
**git config --global user.email "your@email.com"**
- Show your config:

**git config --list**

**Screen Short:**



The screenshot shows a VS Code terminal window with the following content:

```
haiderali@Haider-pc:~/OperatingSystem/Home_Tasks/Task1$ git config --list
user.name=HaiderAli2027
user.email=haiderali223208@gmail.com
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
remote.origin.url=https://github.com/HaiderAli2027/OperatingSystem.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
haiderali@Haider-pc:~/OperatingSystem/Home_Tasks/Task1$
```

## 2. Generate SSH Keys

- Run:

**ssh-keygen -t ed25519**

- Copy the public key:

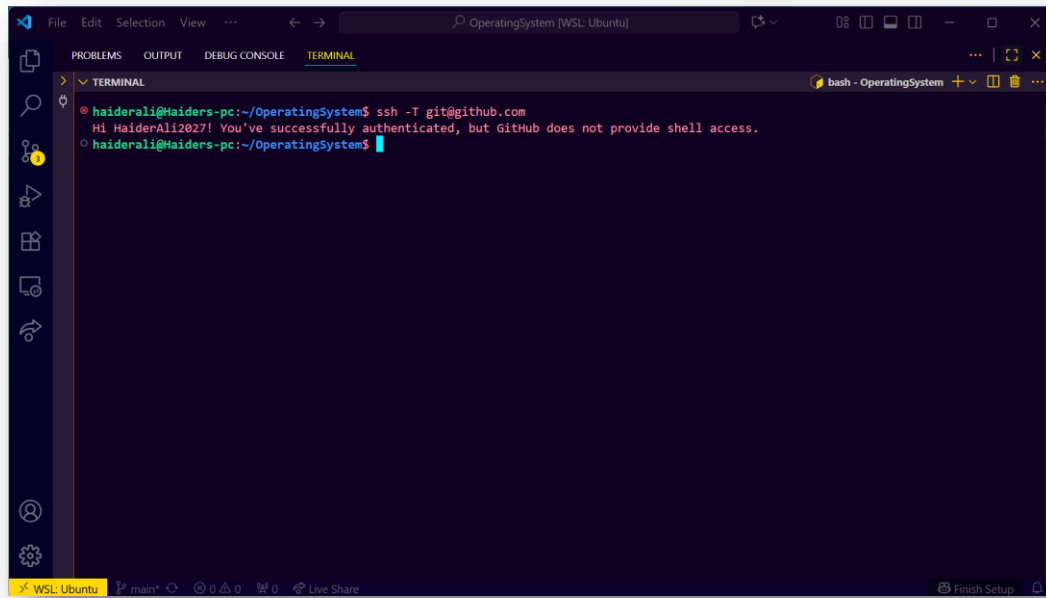
**cat ~/.ssh/id\_ed25519.pub**

Add this key to your GitHub account under Settings → SSH and GPG keys.

## 3. Test Connection

**Ssh -T git@github.com**

**Screen Short:**



## Part C: C Programming Environment & Practice

### Step 1: Install Build Tools

Before writing C programs, install the build-essential package which contains and other tools required for compiling.

**Run:** `sudo apt install build-essential`

Verify installation by checking the version of gcc :

**gcc --version**

**Screen Short:**

```
Command Prompt
C:\Users\LENOVO>gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=C:/msys64/ucrt64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/lto-wrapper.exe
Target: x86_64-w64-mingw32
Configured with: ../gcc-13.2.0/configure --prefix=/ucrt64 --with-local-prefix=/ucrt64/local --build=x86_64-w64-mingw32 --host=x86_64-w64-mingw32 --target=x86_64-w64-mingw32 --with-native-system-header-dir=/ucrt64/include --libexecdir=/ucrt64/lib --enable-bootstrap --enable-checking=release --with-arch=nocona --with-tune=generic --enable-languages=c,lto,c++,fortran,ada,objc,obj-c++,jit --enable-shared --enable-static --enable-libatomic --enable-threads=posix --enable-graphite --enable-fully-dynamic-string --enable-libstdcxx-fil esystem-ts --enable-libstdcxx-time --disable-libstdcxx-pch --enable-lto --enable-libgomp --disable-libssp --disable-multilib --disabl e-rpath --disable-win32-registry --disable-nls --disable-werror --disable-symvers --with-libiconv --with-system-zlib --with-gmp=/ucrt 64 --with-mpfr=/ucrt64 --with-mpc=/ucrt64 --with-isl=/ucrt64 --with-pkgversion='Rev3, Built by MSYS2 project' --with-bugurl=https://g ithub.com/msys2/MINGW-packages/issues --with-gnu-as --with-gnu-ld --disable-libstdcxx-debug --with-boot-ldflags=-static-libstdc++ --w ith-stapel-ldflags=-static-libstdc++
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 13.2.0 (Rev3, Built by MSYS2 project)

C:\Users\LENOVO>
```

## Step 2: How to run a C Program

1. First write a C program in a file with .c extension.
2. Compile the file using: **gcc filename.c -o filename.out**
3. Execute it using: **./filename.out**

## Step 3: Write a C Program

Write a simple C program of your choice. It can be a Hello World program or any other.

