📘 C Development Environment - Step-by-Step Cookbook

System Programming 1 - School of Computer Science, Ariel University

Author: Lior Brown
Target setup: Windows 11 + WSL 2 + Ubuntu 24.04.1 LTS
Goal: Compile, run, and debug C with gcc/gdb + VS Code (WSL)

Step 1 - Install WSL 🪟

1. Open **PowerShell as Administrator**.
2. Run:

   ```
   wsl --install
   ```

3. Restart your computer.
4. Ensure WSL 2 is default:

   ```
   wsl --set-default-version 2
   ```

Step 2 - Install Ubuntu 24.04 LTS 🐧

1. Open **Microsoft Store → Ubuntu 24.04 LTS → Install**.
2. Launch Ubuntu from the Start menu.
3. Create a username & password (remember them).
4. Verify the version:

   ```
   lsb_release -a
   ```

   Should show: `Description: Ubuntu 24.04.1 LTS`

5. Verify the kernel:

   ```
   uname -r
   ```

Should look like: 5.15.167.x-microsoft-standard-WSL2

<u>Step 3</u> - Install gcc ⚙️

```
sudo apt update
sudo apt install build-essential -y
gcc --version
```

Expected output:

gcc (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0

<u>Step 4</u> - Install gdb 🐞

```
sudo apt install gdb -y
gdb --version
```

Expected output (or similar):

GNU gdb (Ubuntu 15.0.50.20240403-0ubuntu1) 15.0.50.20240403-git

---

<u>Step 5</u> - Install VS Code 💻

1. Download from **https://code.visualstudio.com**.
2. During setup → check ✅ **"Add to PATH."**
3. Launch VS Code once to confirm installation.

Step 6 - Install Essential Extensions 🧩

In VS Code → **Extensions (Ctrl + Shift + X)** → install:

Extension	ID	Purpose
C/C++ (Microsoft)	ms-vscode.cpptools	IntelliSense + GDB debugging
Remote – WSL (Microsoft)	ms-vscode-remote.remote-wsl	Open projects directly inside Ubuntu

---

Step 7 - Disable AI Autocompletion 🤖🚫

*In this course, we prefer students to **think in C** before the computer does.* AI autocomplete tools (like Copilot) can write correct-looking code that hides real learning gaps.

1. In VS Code, open the **Extensions** sidebar.
2. Search for:
   - **GitHub Copilot**
   - **Intellicode**
   - **AI Toolkit**, or any similar autocomplete extension.
3. If any of these are installed, click the ⚙️ gear → **Disable (Workspace)** or **Uninstall**.
4. To double-check, go to:
   File → Preferences → Settings → Extensions → Copilot
   → uncheck **Enable** or **Autocomplete** options.

You can re-enable them later for advanced projects - but for now, we code with our own brains 🧠.

Step 8 - Connect to WSL 🔧

1. Open **VS Code**.
2. Press **Ctrl + Shift + P** to open the **Command Palette**.

3. Type **“WSL”** and select **“Connect to WSL”** (or **“New WSL Window”**).

This opens a new VS Code window connected to your Ubuntu in WSL.

<u>Step 9</u> - Verify Everything Works ✅

1. In that window, open the **Explorer panel (📁)** → click **“New Folder.”**
2. Name the folder C_Projects and open it.
3. Inside it, create a new file named israel.c.
4. Paste this code:

```
#include <stdio.h>

int main() {

    printf("Am Israel Chay!\\n");

    return 0;

}
```

5. Save the file (Ctrl + s)
6. Open integrated terminal (Ctrl + ~)
7. Type:

```
gcc israel.c -o israel

./ israel
```

Expected output:

Am Israel Chay!

(If you don’t see it — don’t worry, Am Israel is still alive 😄)