

# COMP / ECPE 177 – Computer Networking C Programming In-Class Exercises



*Lab instructions tested for Ubuntu 12.04LTS and Eclipse "Indigo" 3.7.2*

## Getting Started:

1. Start your Virtual Machine program (VirtualBox, VMWare Player, etc..)
2. Start your Linux Virtual Machine
3. Install Compiler tools, Eclipse C Development Tools, and Valgrind debugger
  - o `sudo apt-get update`
  - o `sudo apt-get install build-essential eclipse-cdt valgrind`

## Create a New Eclipse Project:

1. Start Eclipse inside your Linux Virtual Machine
2. Accept the default location for the Eclipse workspace – all of your projects will be created inside this workspace directory
  - a. Check "Use this as the default and don't ask again"
3. Create an new Project: File->New->C Project
  - a. Under Name: Call this project "first\_program"
  - b. Under Executable: Select **Empty Project**
  - c. Under Toolchain: Select **Linux GCC**
  - d. Select Next to advance to the "Select Configurations" wizard
  - e. Select Finish
4. We want to specify some advanced compiler settings. If we were compiling the code at the command line by invoking GCC (GNU C Compiler), we would want to use these custom options: `-std=c99 -Wall -Wextra`
  - o `-std=c99`: Use the more modern C99 Standard
  - o `-Wall -Wextra`: Turning on all warnings forces you to write better, safer C code
5. To set these same options in Eclipse GUI, choose: Project Menu-> Properties -> C/C++ Build (*expand the category*) -> Settings -> Tool Settings -> GCC C Compiler
  - o Warnings tab:
    - Ensure box for `-Wall` is checked
    - Ensure box for `-Wextra` is checked
  - o Miscellaneous tab: Type in `-std=c99`
    - **Append this** to what is already in the field! The completed line should look like this: `-c -fmessage-length=0 -std=c99`
  - o Select OK

6. Add a new folder called "src" to put your source code in
  - a. Right click on your project name ("first\_program") in the Project Explorer region of the screen
  - b. Choose New->Source Folder
  - c. Enter the folder name: src
  - d. Click Finish
7. Add a new source code file called "first\_program.c" inside the "src" folder
  - a. Right click on the new src folder that appeared in Project Explorer (inside your project)
  - b. Choose New->Source File
  - c. Enter the file name: first\_program.c
  - d. Click finish
8. Enter the source code shown below into the new file.
9. **Save the file** – Eclipse does not auto-save your work before compiling!
10. Compile the project
  - a. Right-click on your project name ("first\_program") in the Project Explorer
  - b. Choose Build
  - c. *You should see compiler success messages appear in the Console pane at the bottom*
11. Run the project:
  - a. Right-click on your project name ("first\_program") in the Project Explorer
  - b. Choose Run As -> C/C++ Program
  - c. *You should see program output appear in the Console pane at the bottom. You can type into this pane to give input to your program.*
12. **Show me the program output.**

```
#include <stdio.h>

int main()
{
    char word[256];
    printf("Tutorial demo program\n");

    for(int i=0; i<15; i++) {
        printf("Value of i: %i\n", i);
    }

    printf("Enter a word: ");
    scanf("%s", word);
    printf("The word is %s\n", word);

    return (0);
}
```

## Debugging:

1. Eclipse has a graphical debugger similar to Visual Studio.
2. Switch to the debugging mode: Window menu->Open Perspective->Debug
3. Perform the following actions in the demo program
  - a. Add a breakpoint: Select the line, and choose Run menu->Toggle Breakpoint
  - b. Run to the breakpoint:
    - i. Choose Run menu->Debug to start your program at beginning
    - ii. Choose Run menu->Resume to let your program run to breakpoint
  - c. Step into, step over, ... using toolbar buttons
  - d. Switch out of the debugging mode (back to coding!)
    - i. Choose Run menu->Terminate to stop debugger
    - ii. Choose Window menu->Open Perspective->C/C++  
*(There are also toolbar buttons to do this)*
4. **Show me your program running in the debugger**

## Command-Line Usage:

*Not familiar with the command line? This ECPE 170 may be helpful for the basics:*

<http://ecs-network.serv.pacific.edu/ecpe-170/labs/lab-linux>

1. Open a command prompt
2. Navigate (using "cd" to first reach your workspace directory, then the project directory inside, and finally the Debug directory inside that.)
3. Compile your program at the command line using this command:  
make
  - a. This runs the exact same compilation process that Eclipse uses
4. Run your program at the command line:  
./your-program-name
5. Use a text editor (GEdit? Emacs? Nano? Eclipse raw text file?) to create a test case file for your program
  - a. Filename = test\_case\_1.txt (Save it in same directory as program)
  - b. Put a single word in your test case file – whatever you want to test your program with
6. Run your program at the command line, and redirect the file into your program as "standard input" (i.e. equivalent to typing at the console)  
./your-program-name < test\_case\_1.txt
7. **Show me your program running at the command line using output redirection**

## Homework #3:

1. **Start working on the third homework**
2. *Not sure where to begin?* The demo program provided on the homework description page (under the resources section) would be a good start. It dynamically allocates a 2D array, initializes it, prints it, and frees it later.