

CS341L: Computer Architecture



- Syllabus <https://www.unm.edu/~sbruno1/cs341l>

CS341L: Linux and Labs



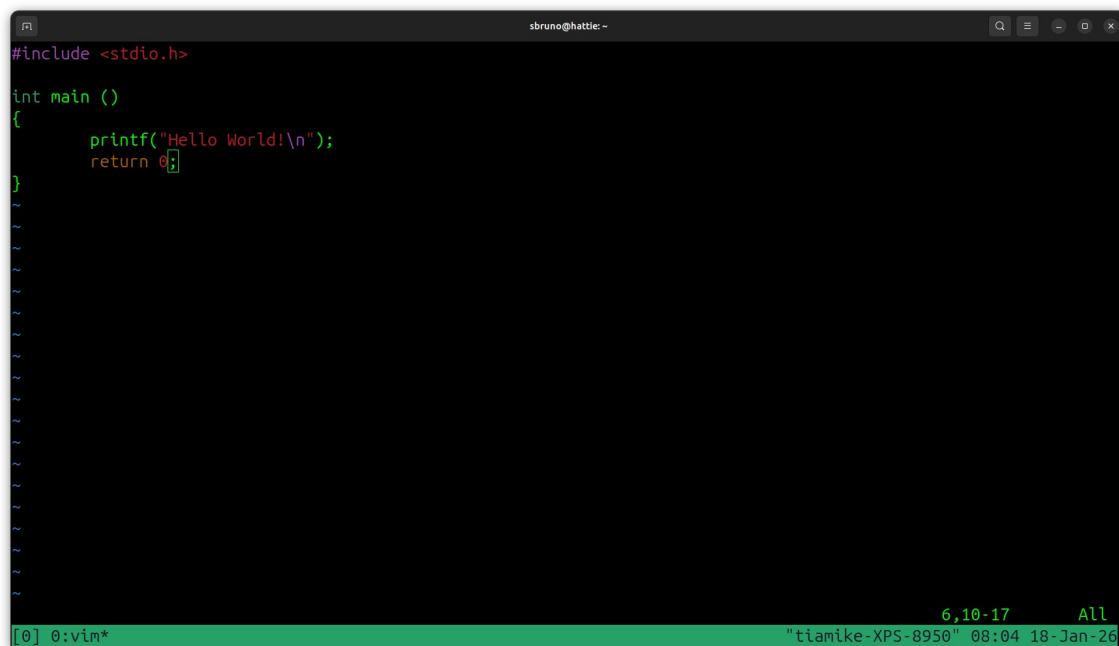
- Terminals
- Editor
- Shells
- Compilers
- Makefiles

```
#include <stdio.h>

int main ()
{
    printf("Hello World!\n");
    return 0;
}
~
~
```

CS341L: Terminal

- Gnome Terminal
- Power Shell
- Mac Terminal
- Xterm
- Tmux/Screen
- WSL

A screenshot of a terminal window with a dark background. The terminal title bar shows 'sbruno@hattie: ~'. The code displayed is a C program for 'Hello World' using printf. The code is:

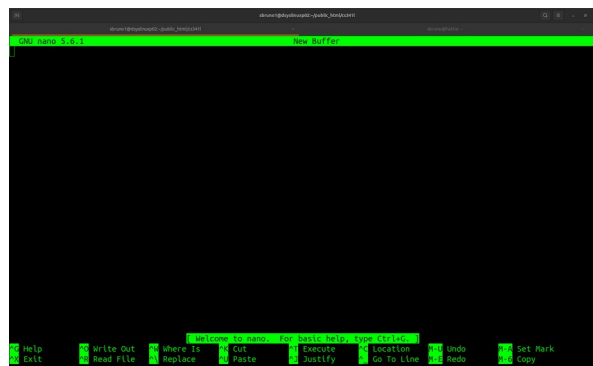
```
#include <stdio.h>

int main ()
{
    printf("Hello World!\n");
    return 0;
}
```

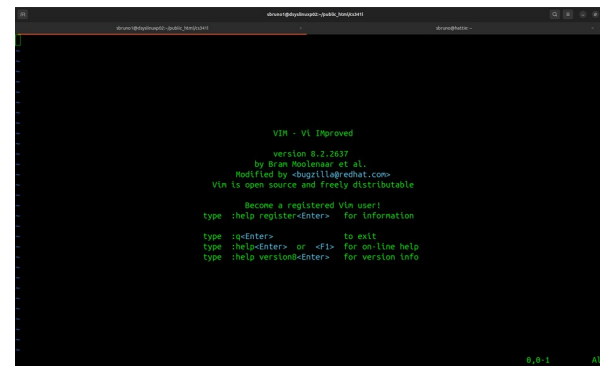
 The vim status line at the bottom shows '[0] 0:vim*' on the left and '6,10-17 All' on the right. The system bar at the very bottom shows '"tiamike-XPS-8950" 08:04 18-Jan-26'.

CS341L: Editors

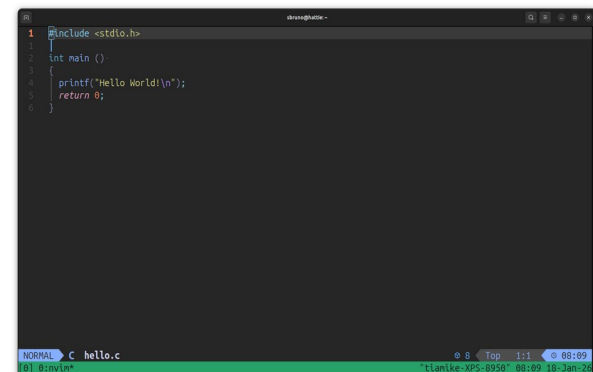
- Nano
- Joe
- VI/VIM
- NeoVim
- VSCode
- Sublime
- EMACS



A screenshot of the Nano text editor running in a terminal window. The title bar shows 'Nano: 5.4.1'. The interface has a green header bar with 'New Buffer'. A green status bar at the bottom displays 'Welcome to Nano', a list of shortcuts (e.g., Ctrl+W for Write Out, Ctrl+O for Open File), and 'Set Mark Copy'.



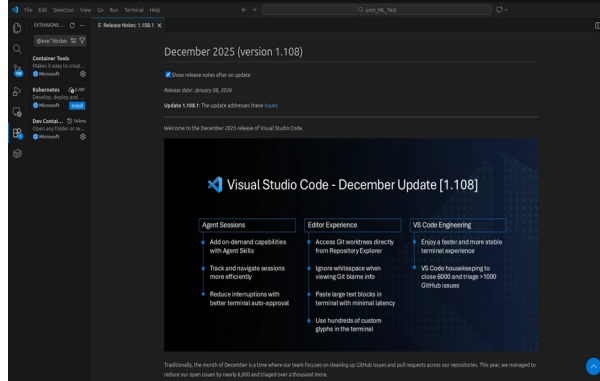
A screenshot of the Vim text editor showing its startup screen. It displays 'VIM - VI Improved', version 8.2.2637, and credits to Bram Moolenaar. It also states that Vim is open source and freely distributable. Instructions for becoming a registered user and using help are provided.



A screenshot of the Visual Studio Code editor. The editor window shows a C program named 'hello.c' with the following code:

```
1 #include <stdio.h>
2
3 int main ()
4 {
5     printf("Hello World!\n");
6     return 0;
7 }
```

The status bar at the bottom indicates the file is 'hello.c' and the editor is in 'NORMAL' mode.



CS341L: Shells and Manual Pages



- Bash
- ZSH
- KSH
- CSH
- SH

```
sbruno@hattie: ~  
BASH(1) General Commands Manual BASH(1)  
  
NAME  
    bash - GNU Bourne-Again SHell  
  
SYNOPSIS  
    bash [options] [command_string | file]  
  
COPYRIGHT  
    Bash is Copyright (C) 1989-2022 by the Free Software Foundation, Inc.  
  
DESCRIPTION  
    Bash is an sh-compatible command language interpreter that executes commands read from the standard input or from a file. Bash also incorporates useful features from the Korn and C shells (ksh and csh).  
  
    Bash is intended to be a conformant implementation of the Shell and Utilities portion of the IEEE POSIX specification (IEEE Standard 1003.1). Bash can be configured to be POSIX-conformant by default.  
  
OPTIONS  
    All of the single-character shell options documented in the description of the set builtin command, including -o, can be used as options when the shell is invoked. In addition, bash interprets the following options when it is invoked:  
  
    -c      If the -c option is present, then commands are read from the first non-option argument command_string. If there are arguments after the command_string, the first argument is assigned to $0 and any remaining arguments are assigned to the positional parameters. The assignment  
Manual page bash(1) line 1 (press h for help or q to quit)  
[0] 0: bash* "tiamike-XPS-8950" 08:17 18-Jan-26
```

CS341L: Compilers



- GCC

```
sbruno@hattie:~$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-linux-gnu/13/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none:amdgc-n-amdhsa
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-linux-gnu
Configured with: ../src/configure -v --with-pkgversion='Ubuntu 13.3.0-6ubuntu2~24.04' --with-bugurl=file:///usr/share/doc/gcc-13/README.Bugs --enable-languages=c,ada,c++,go,d,fortran,objc,obj-c++,m2 --prefix=/usr --with-gcc-major-version-only --program-suffix=-13 --program-prefix=x86_64-linux-gnu- --enable-shared --enable-linker-build-id --libexecdir=/usr/libexec --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --enable-bootstrap --enable-clocale=gnu --enable-libstdcxx-debug --enable-libstdcxx-time=yes --with-default-libstdcxx-abi=new --enable-libstdcxx-backtrace --enable-gnu-unique-object --disable-vtable-verify --enable-plugin --enable-default-pie --with-system-zlib --enable-libphobos-checking=release --with-target-system-zlib=auto --enable-objc-gc=auto --enable-multiarch --disable-werror --enable-cet --with-arch=32=i686 --with-abi=m64 --with-multilib-list=m32,m64,mx32 --enable-multilib --with-tune=generic --enable-offload-targets=nvptx-none=/build/gcc-13-fG75Ri/gcc-13-13.3.0/debian/tmp-nvptx/usr,amdgc-n-amdhsa=/build/gcc-13-fG75Ri/gcc-13-13.3.0/debian/tmp-gcn/usr --enable-offload-defaulted --without-cuda-driver --enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu --with-build-config=bootstrap-lto-lean --enable-link-serialization=2
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 13.3.0 (Ubuntu 13.3.0-6ubuntu2~24.04)
```

- CLANG/LLVM

```
sbruno@hattie:~$ gcc hello.c -o hello_gcc
sbruno@hattie:~$ clang hello.c -o hello_clang
sbruno@hattie:~$ ls -l hello_clang hello_gcc
-rwxrwxr-x 1 sbruno sbruno 16008 Jan 18 08:24 hello_clang
-rwxrwxr-x 1 sbruno sbruno 15960 Jan 18 08:23 hello_gcc
sbruno@hattie:~$
```

CS341L: Makefiles

- Makefile structure
- Variables
- Targets
- PHONY Targets

```
CC=gcc

SOURCE=hello.c

PROGRAM=hello

${PROGRAM}: ${SOURCE}
    ${CC} ${SOURCE} -o ${PROGRAM}

.PHONY: clean

clean:
    rm ${PROGRAM}

~
```