

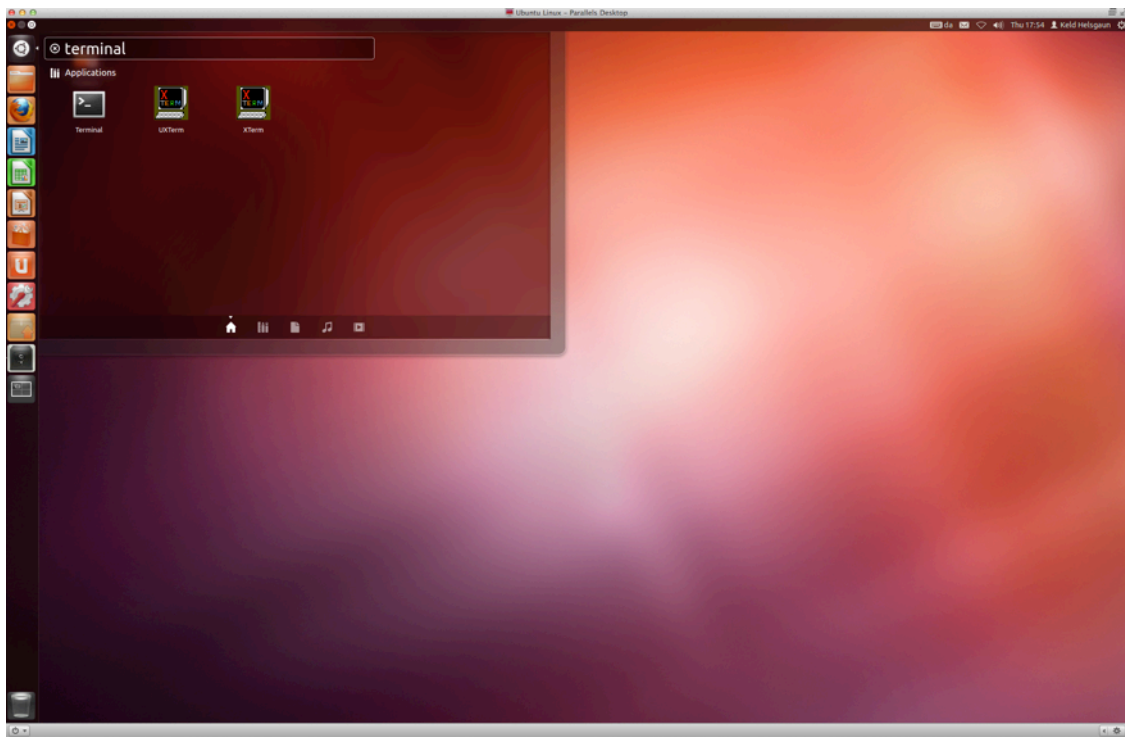
How to Compile and Run a C Program on Ubuntu Linux

Keld Helsgaun
Roskilde University, February 2013

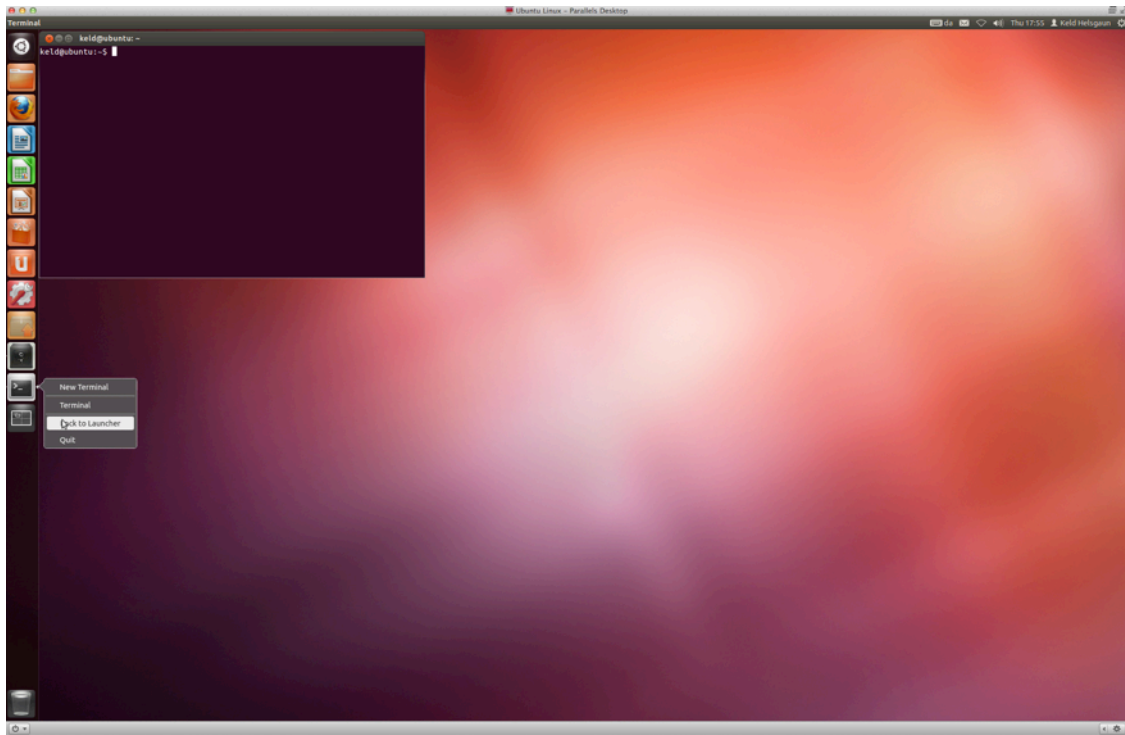
This document shows how to compile and run a C program on Ubuntu Linux using the gcc compiler.

Step 1. Open up a terminal

Search for the terminal application in the Dash tool (located as the topmost item in the Launcher). Open up a terminal by clicking on the icon.



For ease of future access to the terminal application, right click its icon in the Launcher and select “Lock to Launcher”.



Step 2. Use a text editor to create the C source code.

Type the command

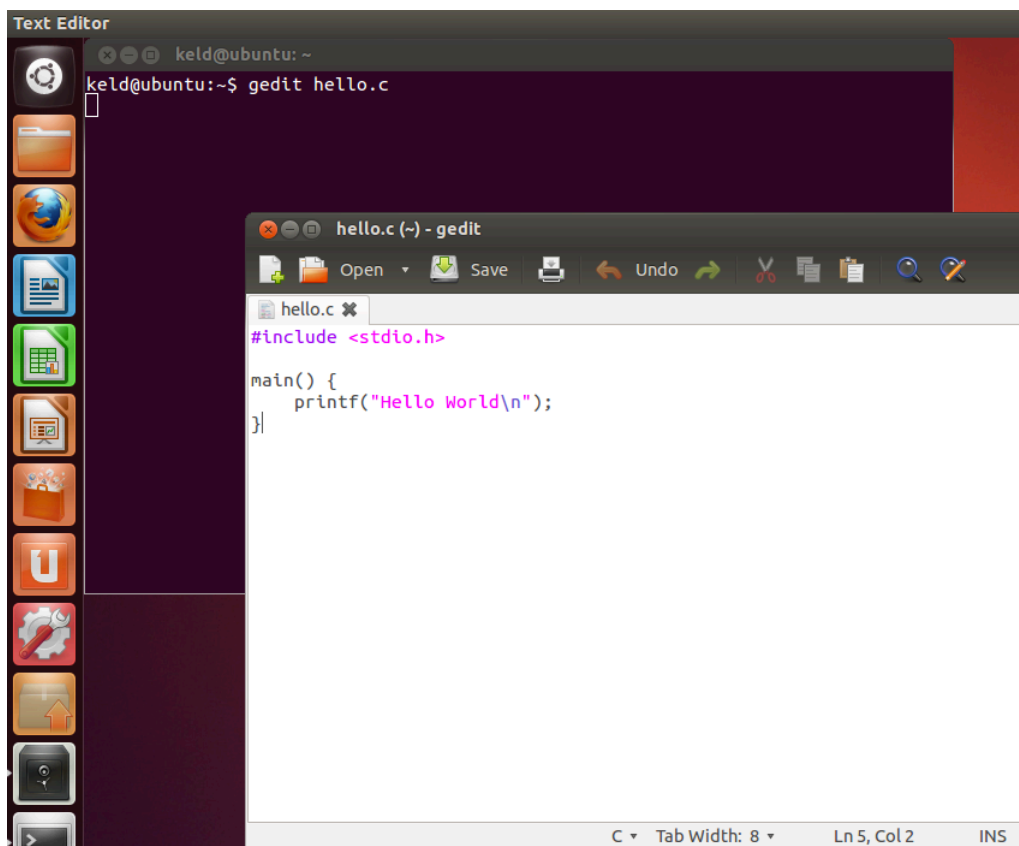
```
gedit hello.c
```

and enter the C source code below:

```
#include <stdio.h>

main() {
    printf("Hello World\n");
}
```

Close the editor window.

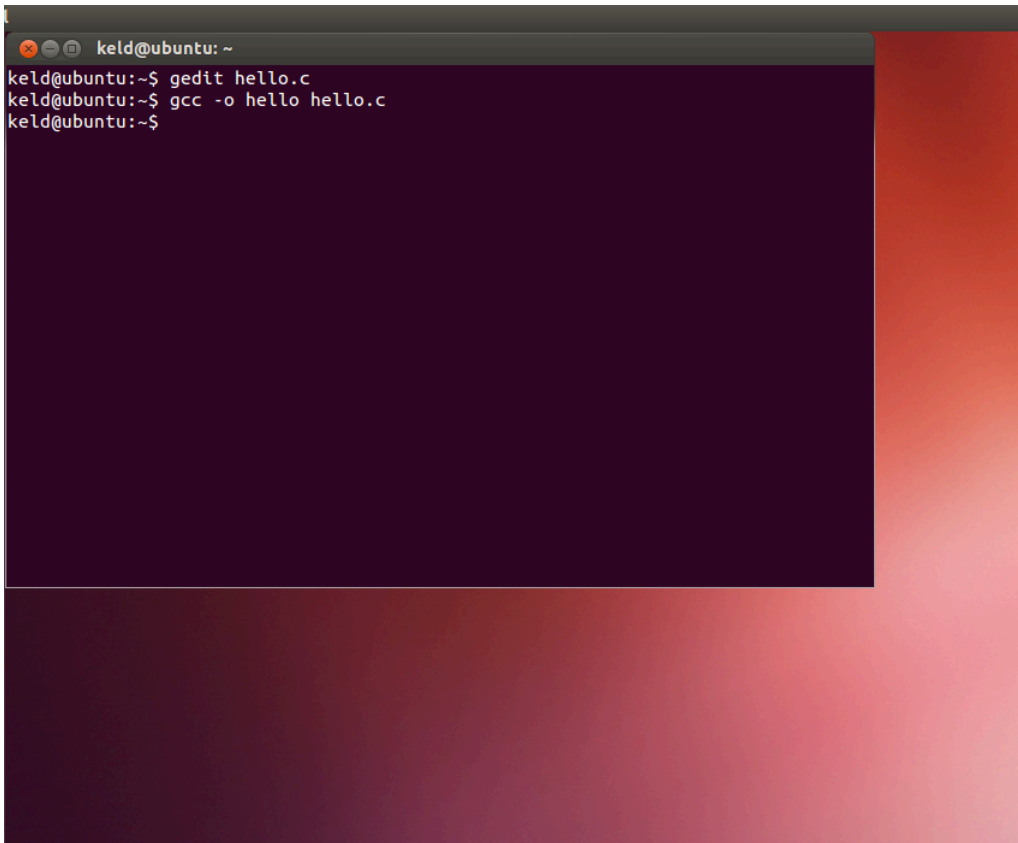


Step 3. Compile the program.

Type the command

```
gcc -o hello hello.c
```

This command will invoke the GNU C compiler to compile the file `hello.c` and output (-o) the result to an executable called `hello`.

A terminal window screenshot showing the execution of the gcc command. The terminal title is 'keld@ubuntu: ~'. The command history shows: 'keld@ubuntu:~\$ gedit hello.c', 'keld@ubuntu:~\$ gcc -o hello hello.c', and 'keld@ubuntu:~\$'. The terminal background is dark purple with a red-to-white gradient on the right side.

```
keld@ubuntu: ~
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$
```

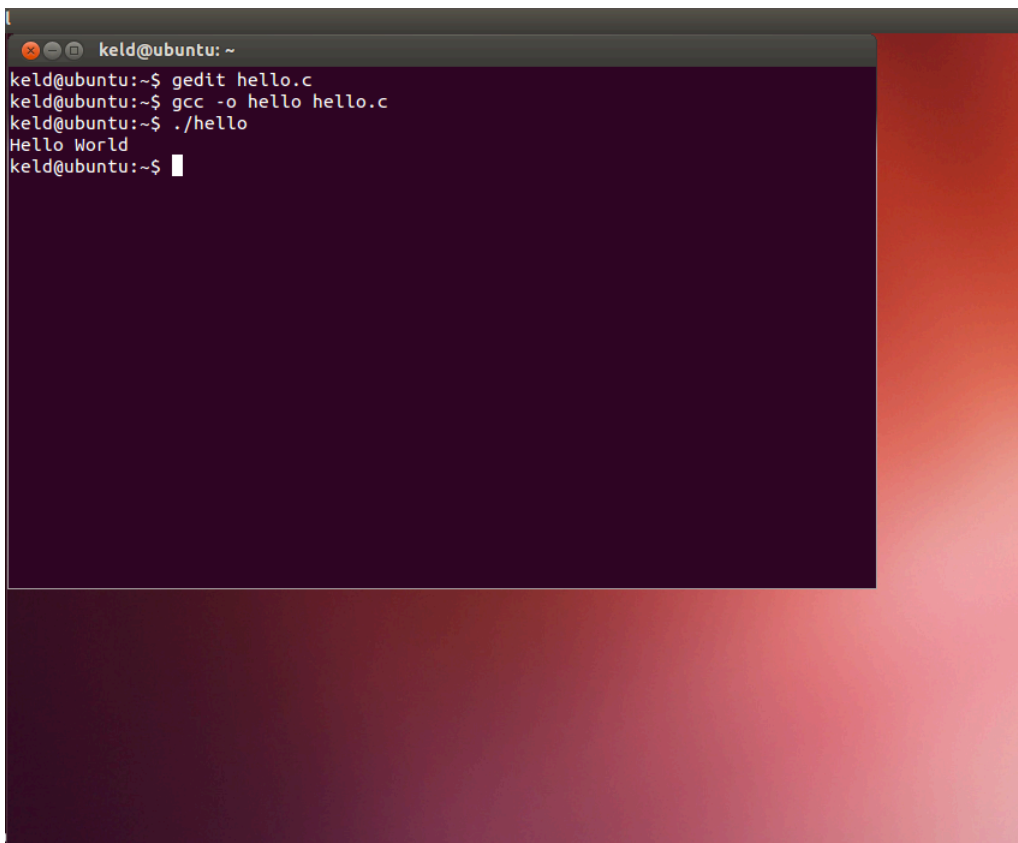
Step 4. Execute the program.

Type the command

```
./hello
```

This should result in the output

```
Hello World
```

A terminal window titled 'keld@ubuntu: ~' with a dark purple background. The terminal shows the following sequence of commands and output:

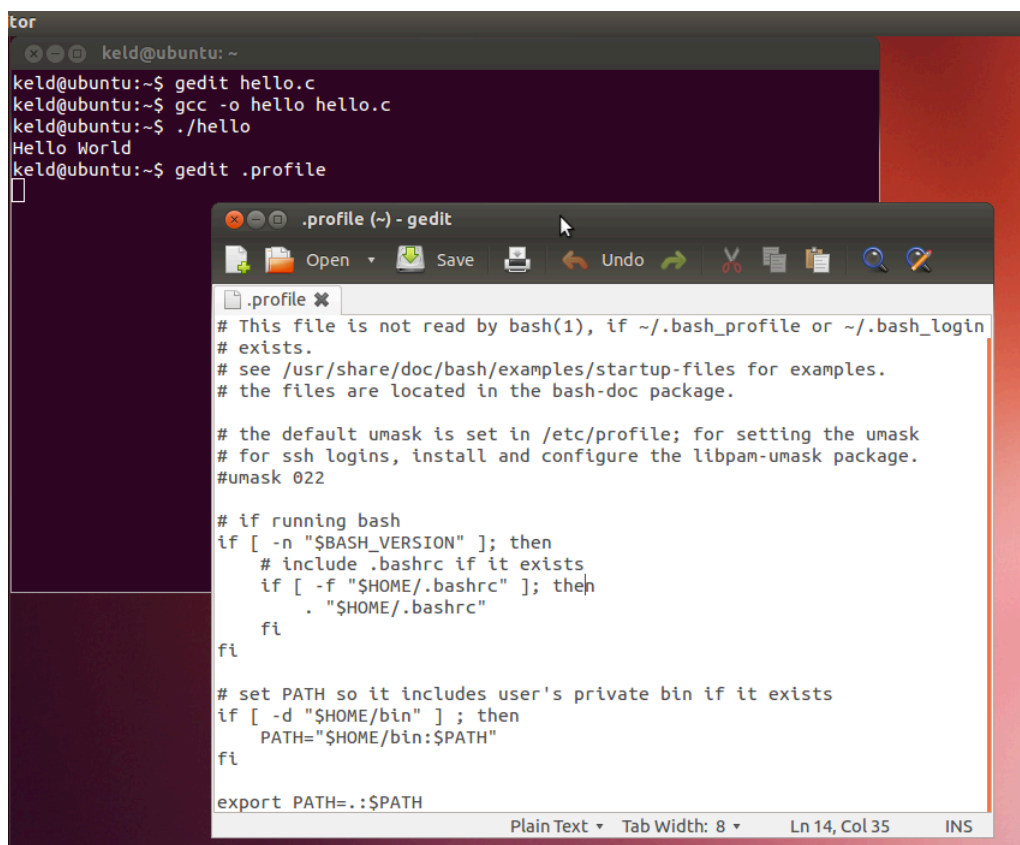
```
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$
```

Optional step

In order to avoid the ./ prefix each time a program is to be executed, insert the following as the last line in the file `.profile` (located in your home folder):

```
export PATH=.:$PATH
```

This step needs only to be done once.



The image shows a terminal window and a gedit editor. The terminal window displays the following commands and output:

```
tor
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$ gedit .profile
```

The gedit editor window shows the content of the `.profile` file:

```
#!/bin/bash
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
#umask 022

# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ]; then
    PATH="$HOME/bin:$PATH"
fi

export PATH=.:$PATH
```

```
keld@ubuntu: ~
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$ gedit .profile
keld@ubuntu:~$ source .profile
keld@ubuntu:~$
```

```
keld@ubuntu: ~
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$ gedit .profile
keld@ubuntu:~$ source .profile
keld@ubuntu:~$ hello
Hello World
keld@ubuntu:~$
```