Preamble to Linux and C

Gilles Audemard — Ibn Sina School for Computer Science

May 2014

This document is a foreword to the two courses: *"Problem Solving in C"* and *"Programming with Data Structures."* The document aims to introduce the development environment used in these two courses: The Linux operating system and the C programming language. This document must be studied and mastered before attending the Ibn Sina School.

The document is being made available to a short list of applicants to the Ibn Sina School (summer of 2014). Applicants are expected to study this document and try to finish as many of the suggested readings and exercises as they can, in preparation for their technical interview with the School.

The Linux environment

Linux is an operating system, like Windows XP and MAC OSX. Linux is based on the UNIX operating system, and was introduced around 1960 by AT&T labs in the USA. Appendix A discusses the reasons for choosing Linux as the operating system for the above courses.

You will need to have basic familiarity with Linux before you join the Ibn Sina School. Linux will be installed on the computers you will use at the School. However, to become familiar with Linux, you will need to practice on one of the computers at your current university which has Linux installed on it. Alternatively, you can install Linux on your laptop and practice directly there. **This is strongly recommended.** Even though laptops are not strictly required for the Ibn Sina School, they are strongly recommended because you may need to finish your programming assignments during the evening or over the weekend. Therefore, we suggest that you bring a laptop with you to the School, with Linux installed on it.

If you have a MAC, then Linux is already installed on it since the MAC OSX is based on Linux. If you have a Windows laptop, then Appendix B details a step-by-step process for installing Linux on it.

The Linux "File System"

Figure 1 depicts some important parts of the Linux file system and provides some information about the listed directories. Note that the directory separation character (`/`) differs from windows (`\`).

Linux has inherited from UNIX the concept of ownerships and permissions for files. Each file has an owner and 9 different permissions: (read/write/exec)×(owner/group/other). A good introduction to file permissions can be found at: `http://www.linux.org/article/view/file-permissions-chmod`. Please study this document.

The Linux "Terminal"

One can use a Linux system like Windows with the mouse and a graphical user interface (GUI). However, the "Terminal" is essential for programming under Linux. Terminal is similar to CMD (Command Line) in Windows but is much (much) more powerful. Everything can be done with the "Terminal" under Linux. Table 1 gives the main commands that you need to be familiar with in order to deal with files under Linux. These commands are typed in the "Terminal" window. The following tutorials are useful if you are not familiar with linux commands:

- Linux terminal command reference: `http://community.linuxmint.com/tutorial/view/244`
- A 5-minutes essential shell tutorial: `http://community.linuxmint.com/tutorial/view/100`

Please study and practice these tutorials.

cmd	meaning	example
`pwd`	shows the current directory	`pwd`
`cd`	changes the current directory (there are two special directories: `.` which is the current one and `..` which is the father of the current one)	`cd /etc`
`mkdir`	creates a new directory	`mkdir ibnsina`
`rmdir`	removes a directory	
`ls`	lists the files and sub-directories in the current directory (this command takes various options for printing only specific files in the current directory: `-l` is one of these options)	`ls -l`
`cp`	copies a file	`cp fileSRC fileDEST`
`mv`	moves a file to a new place	`mv file1 ibnsina`
`mv`	renames a file	`mv fNAME1 fNAME2`
`rm`	deletes a file	`rm file1`
`chmod`	changes the permissions of a file	`chmod 644 file1`
`man`	prints information on a particular Linux command (think of this as online documentation of Linux)	`man ls`

Table 1: Basic Linux commands that can be typed in "Terminal"

Text editors and other software

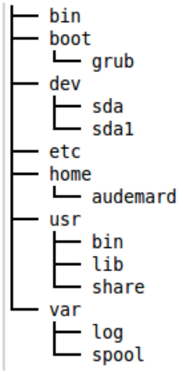
Ibn Sina courses require some software to be installed under Linux:

- A text editor for writing your C programs. The computers you will use at the Ibn Sina School will have *Sublime Text* (`http://www.sublimetext.com/2`) and *Geany* (`http://www.geany.org`) installed. **You need to be comfortable with at least one of these editors.**[1]

- CSH (C shell), which provides an interface to Linux allowing one to control it using C-like commands.

Appendix C provides instructions for installing the above software under LinuxMint (the version of Linux we will use). Other software is also needed: Terminal, the C and C++ language compilers, Makefile and debugger, but these are already pre-installed with the LinuxMint distribution.

[1] If you have a MAC laptop, then *TextWrangler* is a good choice (`http://www.barebones.com/products/textwrangler/`).



```
├── bin
├── boot
│   └── grub
├── dev
│   ├── sda
│   └── sda1
├── etc
├── home
│   └── audemard
├── usr
│   ├── bin
│   ├── lib
│   └── share
└── var
    ├── log
    └── spool
```

Directory tree structure

`/`	Root directory
`/bin`	system executables
`/etc`	Configuration files and loading scripts
`/usr`	The place where the different software is located (looks like Program Files for Windows)
`/dev`	The devices and special files (drives, net)
`/tmp`	Temporary files
`/var`	Data associated with the computer (print files, logs...)
`/boot`	The kernel loaded on boot
`/home`	Users directories

Important directories

Figure 1: The Linux File System

The C language

C is an imperative (procedural) language. It was designed to be compiled using a relatively straightforward compiler, to provide low-level access to memory, to provide language constructs that map efficiently to machine instructions, and to require minimal run-time support. Despite its low-level capabilities, the C language was designed to encourage cross-platform programming.

All admitted students to the Ibn Sina School must have already had a course in either C or C++. Even though C can be thought of as a subset of C++, we have found in the past that students who only took C++ had some difficulty adjusting to C. This is why it is very important that you become familiar with the differences between C and C++. It is even more important that you be able to do the exercises given in this section, which will allow you to have a good start with the first course on "Problem Solving with C." Without being comfortable with these exercises, you will fall behind and may not be able to catch up with your fellow students. This applies to all students, even those who have already finished a course on C.

Differences between C and C++

The following are some notable differences between the two languages:

- C is a procedural language, C++ is an object-oriented language. Hence, there are no classes and sub-classes in C.
- There are no namespaces in C.
- C does not allow the use of reference variables as in C++.
- There are no `new` and `delete` operators in C. Dynamic allocation of memory has to be done with `malloc` and `free` functions in C.
- There are no exceptions in C.
- The standard input and output functions differ in the two languages. C++ uses `cin` and `cout`, whereas C uses `scanf` and `printf`.
- There is no `bool` type in C, false is 0 and true is everything different from 0.

Your first C program under Linux: With step-by-step instructions

The goal of this first exercise is to walk you through the process of writing a C program, compiling it, and then executing it under Linux. The program itself is fully given, together with step-by-step instructions for you to follow. If you still need help beyond these instructions, you may wish to seek help from colleagues who are familiar with C and/or Linux. You may also want to consult the large volume of material that is available on the web for this purpose. For example, to better understand how `printf` works, you can type the following query in Google "printf in C."

The following exercises assume that you have access to Linux. Please read all of the document through it end before you start doing the exercises.

Creating the necessary directories

1. Open a "Terminal" window under Linux.
2. You should already be located in your root directory (`/home/audemard` for me). To find out your current directory, you can type the command `pwd` in terminal.
3. Create a directory ibnsina: `mkdir ibnsina`
4. Go inside it: `cd ibnsina`

```
#ifndef exo1
#define exo1

// Print all divisor of an integer
int allDivisors(int n);

#endif
```

Listing 1: exo1.h

```
#include ”exo1.h”
#include ”stdlib.h”
#include ”stdio.h”

/**
    compute and print all divisors
    @param n: the integer
 */
int allDivisors(int n) {
  int i;
  for(i = 2;i<n/2+1;i++)
    if(n%i==0)
      printf(”%d\n”,i);
}

int main() {
  int nb = −1;
  do {
    printf(”A positive integer please: ”);
    scanf(”%d”,&nb);
  }while(nb<=0);
  printf(”the divisors of %d are:\n”,nb);
  allDivisors(nb);
}
```

Listing 2: exo1.c

Figure 2: The first exercise

5. Create a directory problemSolvingInC, `mkdir problemSolvingInC`, and go inside it, `cd problemSolvingInC`. Note that Linux is case-sensitive, so pay attention to what letters are capitalized.

6. Create a directory preamble and go inside it.

7. You can see the directory structure you have created with the command `tree` ∼. If this command is not installed, install it as discussed in Appendix C.

Typing, compiling and executing the first program

1. Open geany (or any other editor you may wish to use). There are three possibilities for starting geany:

   - Find it in the menu
   - Open it using the launcher: press `Alt+F2`, type geany and enter
   - Type the following command in the terminal:

     ```
     geany &
     ```

     The `&` is used to launch the program as a background process, otherwise, you can not continue to use this terminal.

2. Type the code given in Figure 2. There are two files: `exo1.h` and `exo1.c`. Save them in the right directory (`ibnsina/problemSolvingInC/preamble`).

3. Compile the code by typing the following in terminal:

   ```
   gcc -o exo1 exo1.cc
   ```

   If there is no error, an executable called `exo1` will be created.

4. Test your program by typing the following in terminal:

   ```
   ./exo1
   ```

5. Is it ok?

Your second C program under Linux: Now you need to do more work

The approximation of the square root `x` of a real `a` ($x = \sqrt{a}$) can be computed with the following recursion (called Hero's method):

- $x_0 = a$
- $x_{i+1} = 0.5 \times (x_i + \frac{a}{x_i})$

Your task for this exercise is as follows:

- Create a file `exo2.h` which contains the prototype of a function, called `squareRootApproximation`. This function should take two arguments: a positive real `a` and a positive integer `j`, and return the approximation of the square root of `a` using `j` iterations (i.e., the function should stop and return the value when $i > j$).
- Create the file `exo2.c` which contains the code of this function.
- Create a file `exo2_main.c` which contains the `main` method that:
  - Asks the user to enter the real `a` (be careful, it must be positive) and the integer `j` (also positive).
  - Computes and prints the approximation of the square root using the previous function.
  - Prints the square root using the function. `sqrt`. To use this last function, you need to include the header `math.h`.
- Compile: `gcc -o exo2 exo2.c exo2_main.c`
- Test: `./exo2`

For example, if the function receives `a=20` and `j=4`, then the following iterations are done:

- $x = 20$
- $x = 0.5 \times (20 + 20/20) = 10.5$
- $x = 0.5 \times (10.5 + 20/10.5) = 6.202...$
- $x = 0.5 \times (6.202 + 20/6.202) = 4.713...$
- $x = 0.5 \times (4.713 + 20/4.713) = 4.478...$

Note that $\sqrt{20} = 4.472....$

A Why Linux?

There are two main reasons for choosing Linux as the operating system at the Ibn Sina School:

1. Familiarity with Linux is considered by many as a must for any professional software developer.
2. Many Ibn Sina students appear to have no exposure to Linux, at least during their first two years of college, which can be an obstacle for achieving the School's objectives. For example, some continuity projects will require familiarity with Linux before they can be carried out successfully.

The following list enumerates some of the main advantages and a few drawbacks of Linux (this is somewhat subjective):

- Advantages:
  - Linux is open source: everybody can download sources and modify them
  - Linux is free
  - Linux is stable (no blue screen)
  - Linux is relatively secure (not a lot of viruses, no trojan, ...)
  - Linux is written in C: Good, we are going to learn C!
  - Linux is supported by a lot of software, for music, programming, video and so on
  - Many operations can be automated and executed without a mouse (which can speed up programming)
- Drawbacks:
  - Linux is perhaps lagging behind in its support of video games, but we are not going to play games (expect for the SAT one)
  - Linux may appear difficult to understand and control at first, especially when using Terminal to issue command lines. Yet, many graphical user interfaces exist for Linux (KDE, gnome, unity, ...), which can provide a similar user experience to the one obtained by using Windows.

B Installing Linux on your laptop

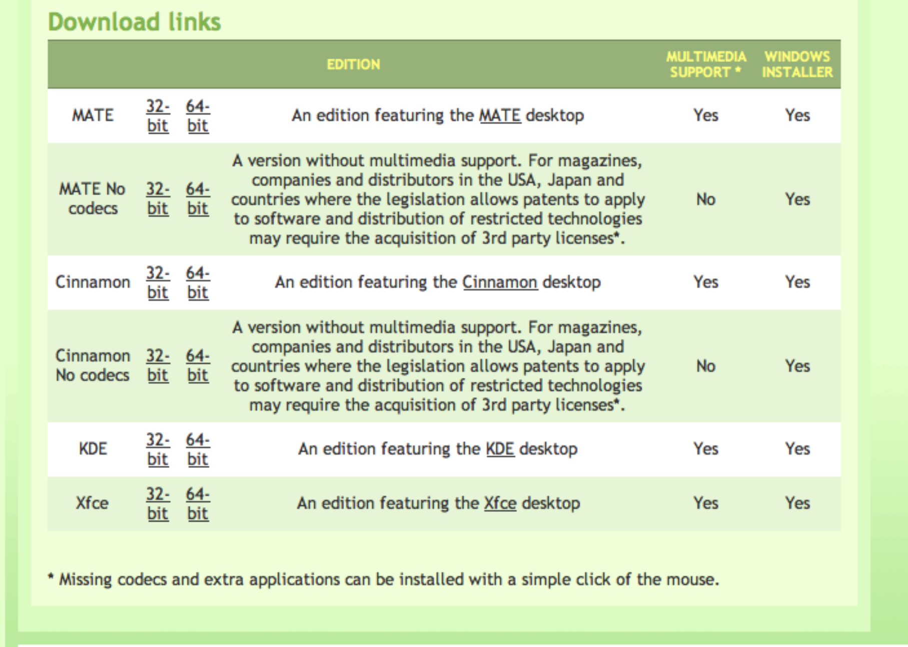
The chosen distribution for the Ibn Sina courses is LinuxMint, Version 14 (called Nadia) (`http://www.linuxmint.com/`). However, all Linux distributions are good alternatives: the system remains the same, with the main differences being in the supported packages (programs) and graphical user interfaces.

The following instructions are for installing LinuxMint on a Windows machine. The installation process requires access to a USB memory stick with 1GB memory. Note that any files on the memory stick will be erased during the installation process.

Please follow the next steps accurately to install LinuxMint:

1. Using a web browser navigate to `http://www.linuxmint.com/download.php`

2. You will find different versions of LinuxMint as shown in the image below. Download the Cinnamon version (third link from the top), choosing 32bit or 64bit depending on your machine.



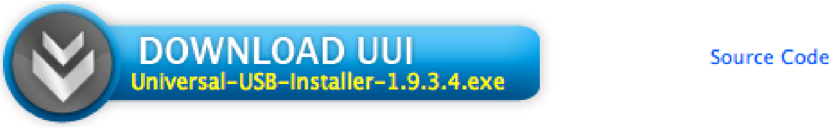
Download links

EDITION				MULTIMEDIA SUPPORT *	WINDOWS INSTALLER
MATE	32-bit	64-bit	An edition featuring the MATE desktop	Yes	Yes
MATE No codecs	32-bit	64-bit	A version without multimedia support. For magazines, companies and distributors in the USA, Japan and countries where the legislation allows patents to apply to software and distribution of restricted technologies may require the acquisition of 3rd party licenses*.	No	Yes
Cinnamon	32-bit	64-bit	An edition featuring the Cinnamon desktop	Yes	Yes
Cinnamon No codecs	32-bit	64-bit	A version without multimedia support. For magazines, companies and distributors in the USA, Japan and countries where the legislation allows patents to apply to software and distribution of restricted technologies may require the acquisition of 3rd party licenses*.	No	Yes
KDE	32-bit	64-bit	An edition featuring the KDE desktop	Yes	Yes
Xfce	32-bit	64-bit	An edition featuring the Xfce desktop	Yes	Yes

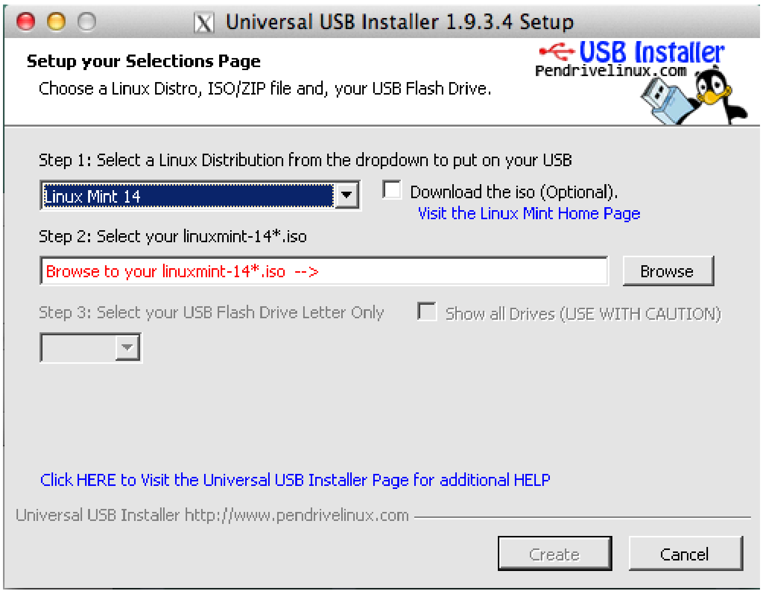
* Missing codecs and extra applications can be installed with a simple click of the mouse.

3. This will take you to another page, where you can select the location to download the file from. In the table labelled "Primary download mirrors", choose a location that is geographically closest to you (any location will actually do). Click on the selected location to download the file, which will have the extension `.iso`. Write down where this file has been downloaded as you will need this information later. Note that this file is large (close to 900 MB) so downloading it may take a while.

4. You will next burn the downloaded file onto your memory stick. For this, you need to download the PenDrive software from `http://www.pendrivelinux.com/universal-usb-installer-easy-as-1-2-3/`. Go to this page, and scroll down until you see the following image. Click on the image to start downloading PenDrive.



5. Double click the downloaded file: `Universal-USB-Installer-1.9.3.4.exe`. Choosing "Yes" and "I Agree" on the next two screens should lead you to the screen below.



Universal USB Installer 1.9.3.4 Setup

Setup your Selections Page

Choose a Linux Distro, ISO/ZIP file and, your USB Flash Drive.

Step 1: Select a Linux Distribution from the dropdown to put on your USB

Linux Mint 14

□ Download the iso (Optional).

Visit the Linux Mint Home Page

Step 2: Select your linuxmint-14*.iso

Browse to your linuxmint-14*.iso -->

Step 3: Select your USB Flash Drive Letter Only

□ Show all Drives (USE WITH CAUTION)

Click HERE to Visit the Universal USB Installer Page for additional HELP

Universal USB Installer http://www.pendrivelinux.com

6. Insert a blank memory stick into your USB port and follow the next three steps according to the above window:

   Step 1: Select LinuxMint 14 from the first drop-down list (scroll further down to find it). Do not check the box "Download the iso."

   Step 2: Click "Browse" and select the `.iso` file you downloaded earlier.

   Step 3: Check the box "Show all Drives," then select the drive corresponding to your memory stick. Caution should be used here as pointing to the incorrect drive may cause all files on that drive to be erased. Do not check the box that says "Format."

   Click "Create" to finish the process. You are now ready to install LinuxMint from the memory stick.

7. Using Windows Explorer, select the drive corresponding to your memory stick. Open that drive. Double click the `mint4win.exe` file to start the installation.

8. You will get three choices on the next screen. Choose the second one: "Install inside Windows."

9. The next screen will ask you to enter a username and a password for the new account. Enter them and click "Install."

10. After the installation process is complete, you will be asked to reboot now or later. Choose "Reboot now" and click "Finish." This will cause the machine to reboot and complete the LinuxMint installation.

11. If all goes well, LinuxMint will be launched next and you will see the "LinuxMint, 14" logo in the middle of your screen.

12. You can now start a terminal by clicking on the black icon in the bottom toolbar: this would be the icon labeled with ">_"

13. If you wish to go back to windows, you will need to restart your machine and then select Windows when given an option between Windows and Linux. You can restart by choosing "Shutdown the computer" from the menu (bottom-left icon in the menu system).

Before you can install additional software, as given below, **you must be connected to the Internet.** To setup your network connection, choose "Menu" from the bottom toolbar, and then type "Network" in the text box at the top of the window that comes up. From there, you can setup your wired or wireless network connection.

C Installing text editors and the C shell

To install the *Geany* text editor, simply type the following command in a terminal:

```
sudo apt-get install geany
```

You can then start the editor by typing

```
geany &
```

To install the Sublime Text editor, type the following commands in a terminal:

```
sudo add-apt-repository ppa:webupd8team/sublime-text-2

sudo apt-get update

sudo apt-get install sublime-text-2-dev
```

You can then start the editor by typing

```
sublime-text-2 &
```

To install the C shell, simply type the following command in a terminal:

```
sudo apt-get install csh
```

To install the `tree` command which helps you see the directory structure, simply type the following command in a terminal:

```
sudo apt-get install tree
```

You can then test it by typing

```
tree ~
```

If you want to learn more about this command and what various options it takes, type

```
man tree
```