Software Systems

Week 1 Lab: Environment Setup and C Programming Revision

Lab By: Jonathan McCarthy

Overview

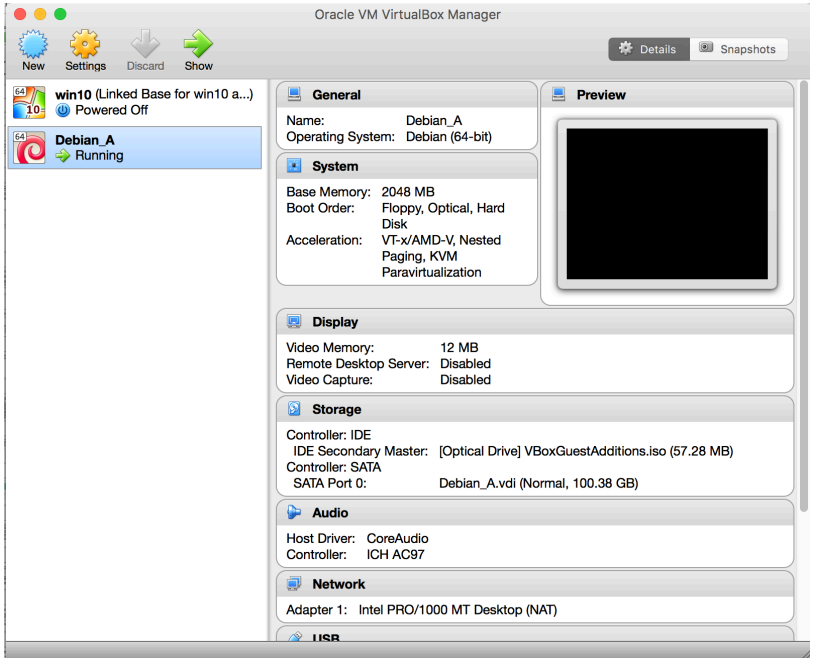
- The following steps should be followed to setup your Linux environment. This is very important that this is completed in Week 1 as this will be used throughout the rest of the module.
- There are a number of different approaches can be taken to get access to a Linux environment.
  - Amazon EC2
  - Virtual Environment
  - Direct Install or Dual Boot
- If you are experiencing any issues with this please contact Jonathan asap (Jonathan.mccarthy@dit.ie)

Install a Linux Distro

- For the lab work this week we will setup a Linux environment using Debian.
- Debian is designed to be a very stable OS. We will need to update the environment and install the required packages to allow us to compile and run C programs.
- If you are running a different OS on your laptop you can install Debian using VirtualBox. This will allow us to install Debian as a virtual OS.

VirtualBox

- If you are going to install Debian as a guest OS on your machine you will need to install VirtualBox.
- Download and install the latest version of VirtualBox
  - https://www.virtualbox.org/wiki/Downloads



- The setup of the VirtualBox environment will be covered later in the lab!!

Debian

- Download the latest version of the Debian Linux ISO:
  - https://www.debian.org/distrib/netinst
  - https://cdimage.debian.org/debian-cd/current/amd64/iso-cd/debian-9.6.0-amd64-netinst.iso
  - Version 9.6.0
- There are different options for installing Debian, if you want to follow a different installation process that's fine.

VirtualBox Setup

- The following steps will detail how to setup VirtualBox for the Debian install.

Steps for VirtualBox

- Create a new OS and give it a unique meaningful name
- Allocate system memory to the virtual OS
- Select create a virtual hard disk (VDI)
- Select dynamically allocated
- Give the system a max hard disk size (eg 100GB)
- Click Create
- In the VirtualBox main screen, click the new system and then click settings

Steps for VirtualBox (continued)

- Click Storage
- Under Controller IDE Click the CD
- On the right of the screen for the optical drive click the CD
- Select the Debian ISO downloaded earlier.
- Click OK
- Click the green arrow to boot the system
- Follow the onscreen instructions to install Debian
- Make sure to select **Gnome**

Setup a C Compiler

- The standard Debian Distro does not contain a C compiler.
  - To test this open a terminal window and run the following command: gcc
- The following slides will bring you through the process.

Steps to install GCC Compiler

- Step 1: Open a Terminal Window
- Step 2: Login as root, type **su** and hit enter
- Step 3: Update the package manager
  - **apt-get update**
- Step 4: **apt-get install build-essential**
- Step 5: Try the **gcc** command

Hello World – Write the program

- Step 1: Open a text editor (**gedit**)
- Step 2: Write the program as save as hello.c

```c
#include <stdio.h>

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

Hello World – Compile the program

- Use gcc to complie the C program in the terminal window.
- Usage: gcc [options] file.c –o outputfile
- To compile the hello.c program. Open a terminal window, navigate to the directory where the C file is.
- To Compile:
  - **gcc hello.c –o hello**

```
File  Edit  View  Search  Terminal  Help
jmccarthy@debianJMC2017:~/Documents/Apps$ gcc helloWorld.c -o hello
jmccarthy@debianJMC2017:~/Documents/Apps$
```

- To Run
  - **./hello**

```
File  Edit  View  Search  Terminal  Help
jmccarthy@debianJMC2017:~/Documents/Apps$ ./hello
Hello, world!
jmccarthy@debianJMC2017:~/Documents/Apps$
```

Create programs for the following:

- Create a C program for each of the following:
  - Calculate area of a rectangle
  - Calculate area of a triangle
  - Calculate area of a circle
- The parameters required for the calculation should be provided in the terminal window when the program is run.

Starter Code

areaOfRectangle.c | main.c | areaOfRectangle.h

```c
#include <stdio.h>
#include "areaOfRectangle.h"


int main(int argc, char **argv) {
   areaOfRectangle(8,9);
}
```

areaOfRectangle.c | main.c | areaOfRectangle.h

```c
#ifndef AREARECTANGLE_H_
#define AREARECTANGLE_H_

int areaOfRectangle(int side1, int side2);

#endif // AREARECTANGLE_H_
```

areaOfRectangle.c | main.c | areaOfRectangle.h

```c
#include <stdio.h>

int areaOfRectangle(int s1, char s2) {
   int area;
   int side1 = s1;
   int side2 = s2;

   printf("Side 1: %d\n", side1);
   printf("Side 2: %d\n", side2);

   area = side1 * side2;

   printf("The area of the rectangle is: %d\n", area);
}
```

The header file allows us to include other C program files in another program

Additional Program Functionality

- For the Rectangle \ Circle \ Triangle area created earlier, create a single program that will allow a user to request an area calculation for a Rectangle \ Circle \ Triangle.
- The parameters should be passed as arguments when the program is run:
  - Eg.
  - ./MyApp Rectangle 10 12
  - ./MyApp Circle 8

Evaluate your solution

- See if your program fulfils the brief as specified in the following:
  - The **UNIX philosophy** by **Doug McIlroy.**
  - Eric Raymond's 17 Unix Rules.

- Do you think the program needs to de refactored to comply with the principles listed above?

To Do:

- Create a C Program to convert a whole number to Roman Numerals.

- **Example:**
- Input (Number): **1079**
- Output (String): **MLXXIX**

Roman Numeral Values

Decimal	Roman Numerals
1	I
4	IV
5	V
9	IX
10	X
40	XL
50	L
90	XC
100	C
400	CD
500	D
900	CM
1000	M

Conversion Example

Example: Convert 2012 to Roman Numerals

Step 1: Identify the highest decimal value that fits within 2012. (1000)

String roman_numeral = “M”;

X = 2012 – 1000;

Step 2: Identify the highest decimal value that fits within variable X. (1000)

roman_numeral = roman_numeral + “M”;

X = X – 1000;

Conversion Example

Example: Convert 2012 to Roman Numerals

Step 3: Identify the highest decimal value that fits within X. (10)

roman_numeral = roman_numeral + “X”;

X = X – 10;

Step 4: Identify the highest decimal value that fits within variable X. (1)

roman_numeral = roman_numeral + “I”;

X = X – 1;

Conversion Example

Example: Convert 2012 to Roman Numerals

Step 5: Identify the highest decimal value that fits within X. (1)

roman_numeral = roman_numeral + “I”;

X = X – 1;

Step 6: X is zero, conversion is complete.

Display roman_numeral.

Logic: For whole number X

1) From the conversion table (slide 4), find the highest whole number that is less than or equal to X.
2) Write down the roman numeral value that you find and subtract its value from X.
3) Repeat steps 1 and 2 until X is zero.

Hint: this can be solved using loops or recursion.

Additional Requirement – all conversions should be written to a text file.

Evaluate your solution

- See if your program fulfils the brief as specified in the following:
  - The **UNIX philosophy** by **Doug McIlroy.**
  - Eric Raymond's 17 Unix Rules.

- Do you think the program needs to de refactored to comply with the principles listed above?

Labwork Upload

Please upload your labwork to Webcourses.