#### **Department of Computer Science**

# COS132 2015 Practical 1



Mr Hannes Janse van Vuuren Copyright © UP 2015 – All rights reserved.

## **Contents**

1	Introduction	2
	1.1 Objectives and Outcomes	
	1.2 Submission	
	1.3 Activities	2
	1.4 Login Details	2
2	The CS Website	3
	2.1 Overview	3
	2.2 Activity: Become familiar with the CS website	4
	2.3 Activity: Book your lab session	4
3	The UP Email Facility	4
•	3.1 Gmail @ Tuks	_
	3.2 Activity: Submit a ticket using Tuks Gmail	
4	The CC Tielest System	6
4	The CS Ticket System 4.1 Overview	6
	4.1 Overview	U
5	Linux	7
	5.1 Persistent Files	7
	5.2 Text Editors	
	5.3 Activity: basic terminal use	7
6	Archives	9
	6.1 Overview	9
	6.2 Activity: Download and extract an archive	9
7	Source code and Compiling	10
	7.1 Compiling, Testing and Uploading	10
	7.2 Activity: Modify and compile a program	10
	7.3 Activity: Create an archive and upload to Fitchfork	10
	7.4 Optional challenge activity: basic interactivity	11
8	The Discussion Board	12
	8.1 Activity: Introduce yourself on the discussion board	12
9	Finishing Off	12
	9.1 Activity: Log off	12
10	Mark Distribution	13

## 1 Introduction

#### 1.1 Objectives and Outcomes

This practical consists of a series of simple tasks intended to equip students with basic knowledge that will be used throughout this semester and beyond. Students will be introduced to:

- The Computer Science (CS) website
- On-line **bookings** for labs
- On-campus email
- Submitting a **ticket** to the CS ticket system (cos132queries@cs.up.ac.za)
- The **Linux** operating system family
- Using the **terminal** and commands on Linux
- Creating and extracting archives
- Compiling a simple program using the GNU g++ compiler
- Uploading to Fitchfork on the CS website
- Posting a message on the ClickUP discussion board

After completing this practical, students should be familiar with the basics of all of the above. Students should ensure that they are familiar with these concepts covered before starting the next practical.

**Linux**-based operating systems will be used throughout this semester and future Computer Science modules. **Fitchfork** is the Department of Computer Science's automatic assessment system (automarker) for programming assignments and practical exams.

#### 1.2 Submission

This practical should be completed and submitted in-lab within the allocated three hour session.

#### 1.3 Activities

Activities are marked with the icon in this practical assignment. You should pay particular attention to completing the required work of activities.

#### 1.4 Login Details

There are **two** sets of login details that you will use for this module and probably also other modules to follow. It is a good idea to memorize your login details. More information about these and where to get them can be found in the handout notes called **Login 101** 

The first set is your **University of Pretoria (UP) login details** which is used to log into the UP Portal (www.up.ac.za), your Tuks Gmail (upnet.up.ac.za/gmail), and on ClickUP (clickup.up.ac.za).

The second is your **Computer Science (CS) login details** which is used to log into the CS website (cs.up.ac.za) which is explained next.

#### 2 The CS Website

#### 2.1 Overview

The CS site is one of the websites that will be used for this module. It is the central website for all CS modules and you will therefore continue to use it after completing this module.

You can access the site by going to http://www.cs.up.ac.za/courses/COS132. You can also use the shorthand version of the URL cs when you are on campus.

The site will be used to share practical instructions and other course material. It has a facility for uploading your practical assignments and book for certain events. You should visit the CS site regularly for the latest updates and announcements. Use your **CS login details** to access this website. The following screenshot gives a short description on how to use the CS website.



Figure 1: The CS Website

- 1. The **announcements** for the module.
- 2. Course material, notes and practical assignment files you can download.
- 3. **Links** to other important websites for the course.
- 4. The **description** of the course.
- 5. A list of all **lecturers**, tutors and teaching assistants (TAs) for the course. You can identify the course coordinator by the pink hat, the other lecturers by a red hat, the tutors by a blue and the TAs by a grey hat. By clicking on one of the names, you can view a photo and the contact details of the person.
- 6. The details of the COS132 **class representatives** who will be elected later in the semester.
- 7. The facility for uploading your Fitchfork and non-Fitchfork **assignments**. This will only be visible if there are active assignments.
- 8. The facility for **booking** for events. This will only be visible if there are active bookings.
- 9. A list of the **uploads** you have done for all the CS modules.
- 10. Details on all Fitchfork marks and results.
- 11. Details on the practical lab slots you have **booked** for.



#### 2.2 Activity: Become familiar with the CS website

- 1. If you read this you probably downloaded this document from the CS website. If you retrieved it by any other means, please go to the CS website and download it. The steps for doing so are on the last page of the Login 101 handout.
- 2. Login to the CS website to ensure your username and password is working.
- 3. Familiarize yourself with the lecturers by visiting their profiles.



#### 2.3 Activity: Book your lab session

For the practicals to come, you will be required to **book** for a lab session. Students will **not be allowed** to attend a practical session if they didn't book. If you have not already booked your lab session, book a session doesn't clash with your other classes. Since lab space is limited, the booking works on a first-come-first-serve basis.

## 3 The UP Email Facility

## 3.1 Gmail @ Tuks



As a student of the University of Pretoria you are automatically assigned an official Tuks Gmail account. The University sends important course-related correspondence to this email account.

For the next activity you will need to send an email, so first you have to log into your UP email account:

- 1. Use Firefox to go to the following web address: http://upnet.up.ac.za/gmail
- 2. You have to login with your email address and UP password: Your email address is: u followed by your eight-digit student number followed by @tuks.co.za For example: u29145678@tuks.co.za Your password is the same as your password for the UP portal.
- 3. Once you are logged into Gmail, you should see a screen similar to this:

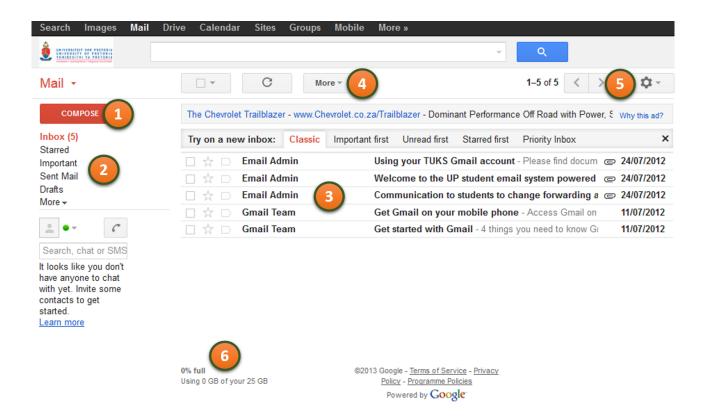


Figure 2: The Gmail web interface

You should know what each marked item or area on the screen is for:

- 1. Write a new email.
- 2. View different categories of mail you received.
- 3. Area where the current email conversation can be read or a list of emails in the current category can be seen.
- 4. Buttons for selecting mails, updating the view, etc.
- 5. The settings menu.
- 6. Gmail storage capacity and use indicator.



## 3.2 Activity: Submit a ticket using Tuks Gmail

1. Send an email to cos132queries@cs.up.ac.za. The subject line should be **exactly** as follows:

COS132 Practical 1 uXXXXXXXX test ticket

where XXXXXXXX is your 8-digit student number. Remember the "u" in front of your student number. Include your name, email address and student number in the message body (order doesn't matter). Make sure you send this email from your Tuks Gmail account and not from a different email account.

You will receive an email response confirming receipt of your email. Allow a couple of minutes for the system to respond.

## 4 The CS Ticket System

#### 4.1 Overview

If you have mark queries, want to book a mentor session or have any subject related question you should email cos132queries@cs.up.ac.za. You should avoid emailing lecturers directly, even if you have a specific question for one of the lecturers. The previously mentioned email address is part of the automated CS ticket system and you will therefore immediately receive an automatic reply.

When sending a email to the ticket system, make sure to follow these rules:

- 1. Always include your name and surname, and student number.
- 2. Provide a short and descriptive subject line.
- 3. Write one email per topic. (eg. don't enquire about quizzes AND a mentor session in the same mail)
- 4. Do not create a new email when continuing correspondence on a query. This helps to keep the mail regarding an issue linked.
- 5. Write a **new** email if you have a new query. (eg. do not reply to a ticket that has been resolved a week ago)
- 6. Your first sentence should be a brief description of your enquiry, followed by a new paragraph explaining your problem.
- 7. Write in a formal style, be respectful and do a spell and grammar check before sending the email.

If the above rules are not followed, *you might not receive any reply* from the lectures. For more information, please read the Netiquette document on ClickUP.

#### 5 Linux

The lab computers run **Arch Linux** with a XFCE graphical desktop environment. In the past, the lab computers used Debian Linux with a GNOME graphical desktop environment.

#### 5.1 Persistent Files

When working on one of the lab computers, the files your create are **not persistent**. That means the moment you log off or restart the computer all the files you created will be deleted. The next time you login the files will be gone.

There is an exception to this rule, namely LinuxShare in your home directory. You can access this folder by using the graphical user interface (GUI) or use the terminal command cd with the location ~/LinuxShare.

It is always a good idea to regularly backup your practical data to an external USB flash drive. However, you will **not** be allowed to use flash drives in the practical semester test and exam.

#### 5.2 Text Editors

In order to write a program in C++ you will need a text editor. There are a number of different editors available, such as SciTE, Vim, gedit and Kate. Your choice of text editor is up to you, but we recommend **SciTE** since it is simple, cross-platform (Windows, Linux, Mac OSX) and available on the lab computers. To open SciTE click on the icon. Alternatively you can use the following terminal command:

scite <file to open> &

Linux filesystems are case sensitive. Therefore to avoid confusion, program names are usually spelled in all lower case. Even though the program is called SciTE, you should use scite as the command. If you don't specify a file to open, SciTE will open and show a blank document. To open a file called file.txt, use the following command:

scite file.txt &

The & at the end means that the command should detach from the terminal, so that you can continue using the terminal for other things. SciTE works whether the & is typed at the end or not.



## 5.3 Activity: basic terminal use

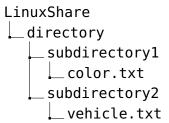
1. Open a terminal and set the current working directory to ~/LinuxShare. This can be done by typing:

#### cd ~/LinuxShare

- cd stands for "change directory"
- ~ (the squiggly character called a 'tilde') stands for "my home directory" which is usually something like /home/u29145678

For the next step you can use the following **commands**:

- You can see what files and directories are in the current directory by typing ls which stands for "list files".
- You can create new directories by typing Mkdir followed by a space followed by the names of the new directories (separated with spaces).
- You can then change into that directory by typing cd followed by a space followed by its name.
- ullet To go back to the parent directory type cd .. The double dot always means 'the directory that contains this current directory'.
- To rename a file or directory type MV followed by a space, its original name, another space and a new name. mv stands for 'move' and it can be used to rename files or to move them to different directories.
- To delete a file you can type rm followed by its name. rm stands for 'remove'. To remove a directory you can type rm -r followed by its name. Be very careful with this command: deleting files is permanent (there is no 'recycle bin' for rm). The file manager's delete command might have a 'recycle bin' function though so you might want to use that instead.
- 2. Produce the following directory structure within LinuxShare:



Where directory, subdirectory1 and subdirectory2 are directories and color.txt and vehicle.txt are text files. You should write text inside color.txt and vehicle.txt using SciTE.

**color.txt** should be a text file that contains a single line with the name of your favourite colour (or a random colour if you don't have a favourite). **vehicle.txt** should contain the name of your favourite vehicle type or brand (or a random brand if you don't have a favourite).

- 3. Show the created stucture to your teaching assistant by issuing the following command at root level in your home folder: tree
- 4. Use the man command (man stands for 'manual') to determine what the following commands do:
  - find
  - cp

#### **General Linux Tip**

Sometimes on the World Wide Web and elsewhere, terminal commands (rather called shell commands) are given with a dollar sign \$ in front. The dollar sign should **not** be typed: it is simply there to indicate that what follows is a shell command. Notice that your shell's prompt also ends in a dollar eg. u29145678@host:~\$. The dollar also indicates that the command will be run as a regular user instead of running as a super-user/administrator (called 'root' on Linux) which is instead indicated by a sharp #

More information on Linux commands and how they work will be given later in the semester.

#### 6 Archives

#### 6.1 Overview

Archives are files that 'contain' other files and directory structures. Archives are often used to combine a collection of files and/or to reduce the total size of the files (via compression). There are a number of different archive types (ZIP, TAR, GZIP, BZ2, RAR, 7ZIP, etc) each with specific a purpose and characteristics. You will receive practical assignments as an archive which you will have to extract in order to use the contained files. You will also be required to create an archive from your practical work and upload it to the online CS system (Fitchfork) for marking.



#### 6.2 Activity: Download and extract an archive

- 1. Download the archive **Practical1.tar.gz** from the CS website. You can download it just as you downloaded this document: the link for it is under "Module Content" on the COS132 CS site. Right click the link, choose 'Save Link As' and choose a directory to save it in (preferably under your LinuxShare directory).
- 2. Open a terminal and navigate (using the cd command) to the directory into which you downloaded the archive.
- 3. Extract the contents of the tar.gz archive with the following command. The archive contains two files: hi.cpp and challenge.cpp tar -f Practical1.tar.gz --extract

You will learn more about extracting and creating archives later in the semester.

## 7 Source code and Compiling

#### 7.1 Compiling, Testing and Uploading

In order for a computer to execute a program, the human-readable code has to be converted into a binary machine-code format that computers can understand. This process is called "compiling". Note that this binary format differs between operating systems, hence C++ code compiled in Windows will not work on Linux and vice versa. We will be using the open-source GNU GCC C++ compiler to compile our programs. The compiler can be invoked by the g++ command. You will learn more about GCC and how to use it in practicals to come.



#### 7.2 Activity: Modify and compile a program

1. Open the file hi.cpp from the archive using a text editor and look at what the C++ code snippet does. Change the program to output the following:

Hi, my student number is XXXXXXXX. I'm busy with a COS132 practical!

Where XXXXXXX represents your student number.

- 2. Make sure that you saved the changes you made to hi.cpp.
- 3. Compile the program by using the following command. You have to open a command terminal and be in the directory where hi.cpp is located when typing the command.

```
g++ hi.cpp -o hi.out
```

You can then run ('execute') the program you just compiled with the following command:

./hi.out



## 7.3 Activity: Create an archive and upload to Fitchfork

1. Use the following command to create a compressed archive (.tar.gz) that contains a copy of your modified hi.cpp source file:

```
tar --create --gzip hi.cpp -f AnswerPrac1.tar.gz
```

- 2. Upload the created archive called AnswerPrac1.tar.gz via the Practical 1 Fitchfork upload link (under "Active Assignments") on the COS132 page of the CS website (http://www.cs.up.ac.za/courses/COS132).
- 3. Navigate to the "My Automarks" section of the website to see your score for this activity (out of six).

- 4. If you wish to try again: modify your program, save the file and make sure it compiles. Then run it to see if it works. Then recreate the archive using the same command given above. Then upload your archive to Fitchfork.
- 5. Ten attempts are allowed for this practical. Every time you upload an archive to Fitchfork, it counts as one attempt. Your previous mark for a Fitchfork upload will be **replaced** every time you upload an attempt.

You will learn more about the way the terminal commands given here work. For now you do not need to understand them completely, just follow the instructions.



## 7.4 Optional challenge activity: basic interactivity

This activity is not mandatory but gives an idea of things you should definitely be capable of before the next practical.

The file challenge.cpp contains a program that performs basic input and processing as well as output. Modify it to create program that asks for the user's name and age, and then outputs a line containing greeting and a statement that the program's age is the entered age plus one.

Compile your program by typing: g++ challenge.cpp -o challenge.out

Then execute it by typing: ./challenge.out

For example, the a user may use the program as follows (user input is shown in **bold**, the rest is program output)

What is your name? **Piet**What is your age? **19**Hello Piet! My age is 20!

Once your program can do this, place your cpp file in an archive:

tar --create --gzip challenge.cpp -f Prac1Challenge.tar.gz

Then upload Prac1Challenge.tar.gz to the Practical 1 Challenge Question upload slot on Fitchfork on the COS132 CS website page. One mark is awarded for the correct input prompts and one mark for the correct output. Since this is a challenge task, these marks do not count towards this practical's mark.

Note: since the above example shows that the user types his answers on the line directly after each question, you should make sure that your program does not output a line break after the question and before user input. The example also shows that the greeting and programs 'age' are output on a single line as two separate sentences, so your program should do the same. Paying attention to the way output is formatted and line breaks are used is **very important**. We try to make our automatic assessment system as lenient as possible, but it is infeasible to prepare it for every possible way students might structure their program output. It will therefore be in your interest to pay attention to examples of program output given in subsequent practical assignments.

#### 8 The Discussion Board

The discussion board on our ClickUP module is there to help students communicate with each another and to ask questions about practicals, Linux and other course-related work. You are not in any way allowed to share your practical work on the discussion board.

You are advised to attentively read the Netiquette document on the ClickUP website before starting to participate on the discussion board. Make sure you understand the gist of our Netiquette rules.

#### A Side Note

We have noticed that students seem not to grasp that the discussion board is mainly to discuss the course material and not for solving administrative issues. Most administrative issues are already answered in the **study guide** and many are more of a personal nature and should be addressed to cos132queries@cs.up.ac.za



#### 8.1 Activity: Introduce yourself on the discussion board

Login on ClickUP and go to the discussion boards. **Locate**<sup>1</sup> the thread named Introduction in the COS132 Admin board. Post a message introducing yourself in one or two sentences in this thread. You can also get bonus marks if you update your picture.

Unlike other relevant posts to the discussion board, this introductory post does **not** contribute to your general discussion board participation mark, but only to the mark for this practical assignment.

## 9 Finishing Off



To avoid abuse of your account, you are advised to log off before leaving the lab. Since other people using the computer afterwards will probably not use Linux, it is usually a good idea to restart the computer. It will then automatically boot to Windows.



## 9.1 Activity: Log off

- 1. Make sure you completed all other activities.
- 2. Log off.

<sup>&</sup>lt;sup>1</sup>Locate means to find – it is there, you have to GO to it.

# 10 Mark Distribution

Activity	Mark
Email	2
Forum post	2
Linux directory structure	5
Fitchfork	6
Total	15