Name: Student Number:

Tut Group:

CORE TASKS – These Tasks are Compulsory

Unix Tutorial Task 1 – Core Tasks [ ]

1. create a directory called **dir1** off your home directory
2. change to **dir1** and print out the path of the current directory
3. create another directory inside directory **dir1** called **dir2**
4. create a zero-length/empty file called **boo** in **dir2**
5. rename the file **boo** to **wibble**
6. create empty files **a.txt**, **b.txt** and **c.txt** in **dir1**
7. archive and zip all **.txt** files in **dir1** into one *tarball* (hint: use **tar** commands with correct options)
8. remove the directory **dir1** and its contents with one command
9. download the file **MyTest.java** from Resources/UNIX-test
10. compile **MyTest.java** using **javac** on the command line
11. write a short (2 line) *Makefile* so you can compile the above program
12. use **Make** to compile the program using your *Makefile*
13. add a rule to the above **Makefile** to remove the program's *class* files
14. demonstrate your removal rule.

OPTIONAL TASKS – You will be asked to do eitehr Task 2 or Task 3

OPTION: Unix Tutorial Task 2 - File Permissions [ ]

1. create a file "**mytest**" (use **touch** to do this)
2. check the permissions on the file
3. set the permission to user = all, group = none, other/world = none; using (r,w,x, +. -)
4. now add *read* permission to those in the same *group*
5. now make sure that the file is readable by *all*, but that no user (including yourself) can delete it.
6. Verify this by trying to remove the file
7. create a directory **test1** and create a file within it called **test2**
8. use a single command to give read and write permission to **test1** and all its descendants

OPTION: Unix Tutorial Task 3 – Job Control & Utilities [ ]

1. open up (or choose an already open and unused) *terminal*
2. start up **vim** from the terminal command line
3. background **vim**
4. bring **vim** into the foreground. close it down normally
5. start and background **vim**; kill **vim** from the command line (use **kill**)
6. type a single command line that will display all the **bash** processes currently running on the system
7. type a single command line that generates a file called **peopledoing.txt** containing the users logged onto the system and what they are doing. Hint: use **ps**
8. type a single line to find all files below directory */usr/include* starting with the letters **iostream** and count the number of files (hint: use **find**)