1. This is a "real" test, which means you must do your own work. It's an open book test, so you can use any resources such as books, your notes, or the computer. However, you must do your own work. This means that you must not ask other students, instructors, acquaintances, paid consultants, Facebook friends, etc. for help. Any violations of the CBC Academic Honesty Policy will result in a failing grade for the course.

    If you use any Internet resources, make sure that you do NOT copy and paste information. You can use the Internet, but you must put all answers in your own words. You will receive no credit for any answers with copied material.

    The test must be completed and turned in by 11:59 on the due date to receive full credit. Because this is the last test and due at the end of the quarter no late tests will be accepted or graded. Tests completed after the due date will not be graded and will receive a grade of 0.

2. Assume you type the command `rm gas???` (gas followed by 3 question marks). What files will be removed?
    a. All files starting with the characters `gas` followed by any three characters.
    b. All files starting with the characters `gas` followed by either 1, 2 or 3 characters
    c. All files starting with the characters `gas` followed by either 0, 1, 2 or 3 characters
    d. All files starting with the characters `gas`

3. Which of the following commands would display all files that start with the characters **song** and end in a number 100-599?
    ```
    a. ls song???
    b. ls song[100-599]
    c. ls song[0-9][0-9][0-9]
    d. ls song[1-5][0-9][0-9]
    ```

4. Assume that somehow you ignored all of my excellent advice and created a file named `j*` How would you delete this file without deleting all files that start with the letter `j`?
    ```
    a.    rm j*
    b.    rm j?
    c.    rm j+
    d.    rm j\*
    ```

5. Which of the following would delete the files `program1.dat` and `program1.out`, and no other files?
    ```
    a.    rm program1.*
    b.    rm program1.[dat, out]
    c.    rm program1.[dat] | [out]
    d.    rm program1.{dat, out}
    e.    rm program1.{dat} | {out}
    ```

6. Assume that your username is ben and you type the following command: `echo \$user is $user` What will be printed on the screen?
    ```
    a.    \$user is $user
    b.    ben is $user
    c.    $user is ben
    d.    ben is ben
    ```

7. True or False. Single quotes will prevent the shell from applying special meaning to all of the following characters: *   ?   ~ $
    a. True
    b. False

8. Assume you type the following command: `cat  >  junk.out`   Identify stdin
    a. Keyboard

     b.   The `cat` command
     c.   The `junk.out` file
     d.   There is no stdin for this command

9. Assume you type the following command: `cat  >  junk.out`   Identify stdout
     a.   Keyboard
     b.   The `cat` command
     c.   The `junk.out` file
     d.   There is no stdout for this command

10.   Assume that you want two C# or C++ programs to share some data. You can write the first program so that it writes it's output to a file, and then write the second program so that it reads the data from this file. However this process can be difficult and time consuming, especially if you don't know how to program in C# or C++. What feature of the UNIX/Linux shell can you use to simplify the process of sharing data between programs?

11. How will `ls >> junk.out` be different from `ls > junk.out`

12. Assume that you type the command `ls -al ~ > homeFiles` and then type the command `date > homeFiles` What will the file `homeFiles` contain?
    a.   The output from the `ls -al` command
    b.   The output from the `date` command
    c.   The output from the `ls -al` command followed by the output from the date command
    d.   The output from the `date` command followed by the output from the ls -al command

13. What can you use, instead of a simple redirect, to temporarily override the noclobber file overwrite protection?
```
a. command >> filename
b. command >\ filename
c. command \> filename
d. command >! filename
e. command !> filename
```

14. Assume that the file phone exists in the current directory. What would you type to display the phone file but change the string (509) to (478) when the file is displayed?
    a.   sed phone | s/(509)/(478)
```
b. cat phone | tr '(509)' '(478)'
c. cat phone | sed /509/478
d. cat phone | sed s/509/478
e. cat phone | sed s/509/478/
```

15. Rewrite and simplify the following sequence of commands using pipes.

```
env > env.out ; sort env.out > sort.out;  more sort.out; rm env.out sort.out
```

16.   Assume that you are currently not in your home directory. What command(s) would you use to find all of the files in your home directory that contain the word "`prompt`"? That is, you want to find files that have the six characters `prompt` as part of their content, not in the filename.

17. Assume you use a redirect to send the output from a command to a file. How do you control whether or not existing files are overwritten or protected?

18. What would the command be to find all accounts that start with **intern** in the password file /etc/passwd, change **intern** to **eng25**, and write all of the account information, that is everything on the found lines, to

a new file named **engineeringAccounts**? Hint - to get full credit you must match only accounts that start with intern, so the string intern must be the first characters on the line.

19. True or False. You can start a command pipeline using the `sed` command.
    a.   True
    b.   False

20. Assume that today is lucky Friday June 13, 2061, and the files below do **not** exist. If I ran the following command what would happen ?

    ```
    touch new_file.`date "+%m%d"`          (These are back quotes and double quotes)
    ```

    a.   You will get an error because the syntax is not correct.
    b.   A file will be created, and it will be named new_file.0613
    c.   A file will be created, and it will be named new_file.date0613'
    d.   A file will be created and it will named new_file.date+%m%d

21. Assume that today is lucky Friday June 13, 2061, and the files below do **not** exist. If I ran the following command what would happen ?

    ```
    touch new_file.'date'          (These are single forward quotes)
    ```

    a.   You will get an error
    b.   A new file will be created and it will be named new_file.0613
    c.   A new file will be created and it will be named new_file.date*
    d.   A new file will be created and it will be named new_file.Friday June 13, 2061

22. Assume that you want to run a command named `birthdayParty`, which is located in the current directory. What would you type to instruct the shell to find this command and run it?
    ```
    a. birthdayParty
    b. /birthdayParty
    c. ~/birthdayParty
    d. ./birthdayParty
    e. /bin/birthdayParty
    f.  run birthdayParty
    ```

23. Which of the following is true?
    a.   The shell will always search the entire filesystem to find any command you type.
    b.   The shell will only search the current directory for commands you type.
    c.   The shell will only find commands if you provide the relative or absolute path to the command.
    d.   The shell will find commands that are in the directories listed in the shell path variable.
    e.   The shell will find commands that are in hash table built from the directories listed in the shell path variable.

24. Which of the following is true?
    a.   There is no way to add the current directory to your shell search path.
    b.   You can add the current directory to your shell path; but if you do it will be ignored.
    c.   You can add the current directory to your shell path; but if you do it should be the first thing in $path to avoid security issues.
    d.   You can add the current directory to your shell path; but if you do it should be the last thing in $path to avoid security issues.

25. Which of the following is NOT true?
    a. You can make changes to your $path permanent by adding the changes to your `.login` file.
    b. You can make changes to your $path permanent by adding the changes to your `.cshrc` file if you're using `tcsh/csh`.
    c. You can test changes to your `.cshrc` or `.login` by using the source command.
    d. The only way to test changes to your `.cshrc` or `.login` files is to log out and log in again.

26. True or False. The shell searches every directory in $path every time you type a command.
    a. True
    b. False

27. What is the name of the command that instructs the shell to rebuild the table containing the commands from the directories listed in your search path?

28. Assume that you want to build an alias. When you type `del` you want the shell to run `rm -i`. What would you type to create this alias?
    ```
    a. alias del rm -i
    b. @alias del rm -i
    c. alias rm -i del
    d. @alias rm -i del
    ```

29. When you login and type the alias command you will see that you already have several aliases; even though you did nothing to create them. How do you permanently remove one of these preset aliases?
    a. These aliases have been created by the System Administrator and there is no way to remove them.
    b. Send a request to the System Adminstrator and ask them to remove the alias for you.
    c. Use the `chalias` command. This allows you to change the aliases that are created when you login.
    d. Edit your `.login` or `.cshrc` and add a command to `unalias` the alias you want to remove.

30. Assume that you want to create an alias named `lm` that runs `'ls -al | more'`. Your alias works if you do not give it a directory as an argument, but if you try and run it on a different directory, e.g. `lm /` it lists the contents of the current directory and then prints an error message. Which of the following would create the alias so that it passes the argument in to the ls command?
    ```
    a. alias lm 'more < ls -al'
    b. alias lm 'ls -al' $1 | 'more'
    c. alias lm 'ls -al !* | more'
    d. alias lm 'ls -al \!* | more'
    ```

31. The `echo` command is one of the shell built in commands. If you create a shell script named `echo` and an alias that is also named `echo`; which command will the shell run when you type `echo` on the command line?
    a. The shell script
    b. The built-in
    c. The alias
    d. None of the above

32. Assume that the last command you typed was `vi .login` Which of the following would tell the shell to repeat that command?
    ```
    a. !!
    b. !-1
    c. !v
    ```
    d. All of the above

33. True or False. Assume that your history contains the following commands. Also assume that you are typing command 59. If you want to repeat command #54, and you typed `!c`, you would get the desired result?

```
53  cd
54  cat  /etc/passwd
55  rm  pwd.txt
56  cp  /etc/passwd  pwd.txt
57  chmod  755  pwd.txt
58  vi  pwd.txt
```

   a.   True
   b.   False

34. How would you change your prompt to have it display the current time (hour and minute) and the *full* path of the current directory? You must change the shell variable "prompt" not the variable "PS1".

35. What does the `#!/bin/sh` or `#!/bin/bash` at the beginning of your shell scripts mean ?

36. What will the following echo command output? You can assume that the date is June 6, 2096

```
echo `date "+%m%y" `
```

   (Note - the outer quotes are back ticks or grave quotes)..
   a. This would cause an error
   b. date +%m%d%y
   c. date 060696
   d. 0696

37. Assume that you have written a shell script named sunset. Within the script you have the command `echo $1`. What will be displayed on the screen by this command if the script is started by typing the following on the command line:

```
./sunset red 8 cloudy
```

   a.   This would cause an error
   b. ./sunset
   c. red
   d. 8
   e. cloudy
   f. sunset red 8 cloudy

38. Assume you are running a shell script which has the command: **echo -n "Enter Last Name: "**
   immediately after this command executed, where would the cursor be ?
                Enter Last Name: A
                B   C

   a.   A
   b.   B
   c.   C
   d.   This would cause an error

39. Assume that you ran a shell script which contained the following code:
```
#!/bin/sh
echo -n "Please enter your favorite foods: "
read fav1 fav2
echo "$fav1"
echo "$fav2"
```
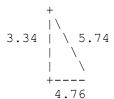
What would the script output if you entered: `ice cream, spam, twinkies`
   a. ice cream
   b. ice cream,
   c. ice cream, spam
   d. ice
   e. none of the above

40. True or False. You can change the value of a command line variable, such as $1, inside a shell program. You do this by simply assigning a new value, the same way you would with any other shell variable.
   a. True
   b. False

41. Write a shell script that uses a loop to build a table of Fahrenheit temperatures and the corresponding Celsius temperatures. You can either write the entire script, or modify an existing script. The script must meet these requirements:

   a. Your name must be included in comments at the top of the file.
   b. The script must ask the user for the starting Fahrenheit temperature, ending Fahrenheit temperature and increment.
   c. To calculate the Celsius temperature use the formula °C = (°F - 32) x 5/9
   d. The table should have the headings "Degrees F" and "Degrees C"
   e. The left hand column of the table should contain Fahrenheit temperature; the right column should contain the corresponding Celsius temperature.
   f. You do NOT need to check for an infinite loop. You can assume that the user will provide reasonable input.

To receive full points your script must be named test2Temp.sh, it must be located in your home directory, and it must have execute permissions of 755.

42. Write a shell script that given two sides of a right triangle, calculates the hypotenuse. The formula for this is $hyp^2 = A^2 + B^2$ Or $hyp = (A^2 + B^2)^{1/2}$ You can either write the entire script, or modify an existing script. Your program must meet the following specifications:

   a. Your name must be included in comments at the top of the file.
   b. It must allow the user to enter the values for A and B, perform the calculation(s), and then print the output. Here are some hints for writing the program.
      • Do the calculations in a couple of steps instead of trying to do them all at once. That is, first multiply A*A, then B*B, then add them together, then do the square root. Check your work at each step by printing the intermediate results.
      • Use the `bc` command to do the square root calculation. It has a sqrt function that can be called with `sqrt($var)`
   c. The output should be a triangle that is labeled appropriately. * Use the | + - and \ characters to print a triangle for the output. It can always be the same triangle, just change the numbers. Do something similar to:

```
      +
      |\
 3.34 | \  5.74
      |  \
      |   \
      +----
       4.76
```

    d.   Use the scale command to limit the number of decimal points for the hypotenuse, but do at least two places after the decimal.

    e.   You can assume that the user will enter numbers, so you don't have to do any error checking of the input.

    f.   After the calculations are complete and the results displayed, the program must ask the user if they want to repeat the process. If the answer in the affirmative then the code must loop and repeat. If the user answers in the negative the code should exit.

To receive full points your script must be named test2Hyp.sh, it must be located in your home directory, and it must have execute permissions of 755.