IB Computer Science

Unit 1

C Programming in Linux

Student_____

Date: _____

Linux Lab 2

In this lab you will run your project code for converting numbers between number systems as well as use your program generating random numbers. You will use a variety of new Linux commands that are available within the Cloud 9 integrated development environment. Be sure to answer all questions and complete all fields in this lab.

Use this summary page to record notes about the Linux commands that you will use during this lab.

Command	Command Notes
bash	
cut	
grep	
head	
hexdump	
od	
sed	
sort	
tail	
tee	
uniq	
wc	

IB Computer Science

Unit 1

C Programming in Linux

Student_____

Date: _____

Step 1 : Generating 25000 Random Numbers

Change your program that generates random numbers to accept a command line argument containing the value of **n**, the number of random numbers to generate. This is a sample program:

```
1)  // Name: Mr. Brennan
2)  // File: my_random.c
3)  // Purpose: This will be a file of functions that use or
4)  //      generate random numbers. It contains a driver main program
5)  //      but can be compiled to only provide library functions.
6)
7)  #define OPT 1   // compile in driver mode
8)
9)  #include <stdio.h>
10)
11) // Function randbetween: return a random number between the two input parameters
12) int randbetween(int min_, int max_) {
13)     // In C the rand function returns and integer
14)     // in other languages it may return a real number
15)     // between 0 and 1, and you would use:
16)     //       int j = (int)((max_-min_+1)*rand()+min_);
17)
18)     int j = rand()%(max_-min_)+min_;
19)     return j;
20)     }
21)
22) #if OPT == 1  // compile it the driver program
23)
24) int main(int argc , char *argv[]) {
25)     int i;
26)     int n;
27)     srand(time(NULL));
28)     if (argc < 2) {
29)        printf("How many random numbers would you like? \n");
30)        scanf(" %d",&n);
31)        }
32)     else {
33)         n = atoi(argv[1]);
34)        }
35)     for (i = 0; i < n; i++)
36)         printf("%d\n",randbetween(1, 32000));
37)     }
38) #endif  // compiling in the driver program
```

Compile and run your program.

a) test your program, generating 10 random numbers: ./a.out 10

b) test your program, generating 1000 random numbers ./a.out 1000

Create data files

c) create a file containing 10,000 random numbers:
./a.out 10000 > random_1000.dat

d) create a file containing 25,000 random numbers:
./a.out 25000 > random_25000.dat

IB Computer Science
Unit 1

C Programming in Linux

Student_____

Date: _____

Step 2 The **tee, wc, and tail Commands**

The tee command is used to display output of a command to the console at the same time that output is copied to a file.

a) Practice using the tee command by creating a file containing 100 random numbers. Issue the command:

./a.out 100 | tee random_100.dat

b) Record the values of the last 5 random numbers that were displayed on the screen:

1.
2.
3.
4.
5.

c) Verify that the file random_100.dat contains 100 lines. Record the output of the wc command:

Command	**Output**
wc random_100.dat	

d) Use the man page to explain what the four fields of the wc command are:

Command	**Output field definitions (explain the output of c) above**
man wc	field 1: field 2: field 3:

e) The **tail** command is used to view the last lines of a text file. 10 lines will be displayed if the number of lines to be displayed is not specified. Record the output of the following tail command:

Command	**Output**
tail –5 random_100.dat	

Does the output from e) match the output from b)? _____

IB Computer Science

Unit 1

C Programming in Linux

Student_____

Date: _____

Step 3 The **uniq, and sort Commands**

a) Record the output of each of the following commands

Command	**Output**
uniq random_1000.dat | wc –l	
sort random_1000.dat | wc –l	
sort random_1000.dat | uniq | wc –l	

b) Use the man pages for uniq and sort to describe/explain the differences in the output from the table above.

c) Make a prediction for the output of the commands in the table below; then execute the commands to see if your predictions were close.

Command	**Predicted Output**	**Actual Output**
uniq random_25000.dat | wc –l		
sort random_25000.dat | wc –l		
sort random_25000.dat | uniq | wc –l		

IB Computer Science
Unit 1

C Programming in Linux

Student_____

Date: _____

Step 4 The **sed Command**

a) Read the **sed** reference http://hwmath.net/Unix/sed_Examples.pdf which as a link from today's agenda page.

b) Use the file random_1000.dat to call your convert program. In this step you will create a batch file to call your program with each of the random numbers.

1. Start by creating a small test file with 8 random numbers.
   `./a.out 8 > random_8.dat`
2. Suppose that the first line in the random_8.dat file contained
   1015
   to call your convert program, you would use the command line
   `./convert -d 1015`
   this means that you need to add the string "./convert –d " at the beginning of each line.
   The sed command can be used to replace /this string/ with /that string/ using the command
   sed 's/this string/that string/' input_file

   In this case, we want to add a string to the beginning of each line. There is a special symbol (^) to use that indicates adding to the beginning of the line (or substituting nothing at the start of a line with something at the start of the line.

   Test the following sed command, which should add the string "convert " to the start of every line of the random_8.dat file:
   `sed 's/^/convert/' random_8.dat`
   a sample line of output should look like
   `convert 1014`

3. Create a batch file to call your convert program using the data from random_8.dat:
   `sed 's/^/.\/convert –d /' random_8.dat > convert_8.sh`

   The sed s command is a substitution command. In this case we are substituting the start of a line with a replacement string, such as sed 's/^/replacement/' . Because out replacement sting "./convert –d ' contains a '/' character, we need to add the '\' character before the '/' like '\/' to tell sed that the '/' is part of the replacement string.

   The output of the sed command is stored in the file convert_8.sh which should not contain 8 command lines to call your convert program.
4. Execute the convert_8.sh file
   Execute the commands in convert_8.sh by starting a new bash shell to run the script:
   `bash convert_8.sh`
   If your **convert** program is in the same directory where your random number generator program is located and all of the test files that you have created for this lab, then your convert_8.sh script should execute. If you get an error that program **convert** can not be found, then either copy the convert_8.sh into the directory where **convert** is located, or find a more create solution to your error.
5. Create a bash script file to call your convert program with 1000 random numbers. Name your batch file convert_1000.sh. Make sure that your workspace is shared with the instructor or you will not get credit for completing this step.

IB Computer Science
Unit 1

C Programming in Linux

Student_____

Date: _____

Step 5 The **grep Command**

The grep command is used to search for strings within files.
Record the output of the following commands (be honest):

Command	**Output**
`ls *.c`	
`ls *.h`	
`grep Name *.c`	
`grep Name *.h`	
`grep  1011 *.dat`	
`grep  2051 *.dat`	

IB Computer Science
Unit 1

C Programming in Linux

Student_____

Date: _____

Step 5 The **hexdump, od, cut and head Commands**

Describe the output of the follow commands

Command	**Command Output Description**
`head -2 random_25000.dat`	
`head -2 random_25000.dat | hexdump`	
`hexdump random_8.dat`	
`hexdump –c random_8.dat`	
`od random_8.dat`	
`od –c random_8.dat`	
`od -c random_8.dat  | cut -f 1 -d ' '`	

The **cut** command above will display the first field of output from the **od** command because the cut command is passed the parameter –f 1. The **cut** command will use a space as a delimiter because of the –d ' ' parameter.

What does the first column of output of the **od** command represent?