

Prep C

Editing Programs

Dr. Madhav Rao

Revision Date: July 1, 2016

Task 1: Linux

Create a directory named *day0* that hangs off your (home) *~/* directory. Create five directories inside the *day0* directory: *q1*, *q2*, *q3*, *q4*, and *q5*. Now, while in the *~/day0* directory, type the system command:

```
du
```

You should see the following output:

```
4  ./q1
4  ./q2
4  ./q3
4  ./q4
4  ./q5
20 .
```

The names must be as shown, but the order and the numbers do not matter.

Task 2: Vim

Move into your *~/day0/q1* directory. Create a file named *textlines.txt* and place in it 375 lines of:

```
C Programming lab is easy if you work hard!
```

followed by 125 lines of:

```
I edit with Vim and I program with C!
```

There should be no spaces at the beginning of the lines and no spaces at the end of the lines. If you go to line 375 in the file, you should see the C Programming line and if you go to line 376 in the file, you should see the Vim and C line. Finally, if you go to line 500 in the file, you should be on the last line in the file. You can use vichecker software available in LMS or web to test your *textlines.txt* file.

Task 3: Vim (10 points)

Move into your *~/day0/q2* directory. Copy the file *textlines.txt* to current directory. Rename this file as *manipulate.txt* in *q2* directory. Do not rename the original file *textlines.txt* in *q1* directory. In *manipulate.txt* file, replace the word *C* to *Python* in all the 500 lines and replace the word *Vim* to *VIM* (Note all letters are upper case in VIM !!) You can use mchecker software available in LMS to test your *manipulate.txt* file.

Task 4: Editing C Program (20 points)

Inside the *day0* directory, move to the *q3* directory. Inside this directory, create a file named *energy.c*. This program will be used to determine the kinetic energy of a moving object based on its mass and velocity. Place the following program into *energy.c*:

```
#include<stdio.h>

int main()
{
    float mass,speed,energy;
    printf("Calculating kinetic energy of a moving object:\n");
    printf("Enter the object's mass in kilograms: ");
    scanf("%f",&mass);
    printf("Enter the object's speed in meters per second: ");
    scanf("%f",&speed);
    energy = mass * speed;
    printf("The object has %.2f joules of energy\n",energy);
    return 0;
}
```

Compile the program *energy.c* as follows:

```
gcc -o energy energy.c
```

Run the program as follows:

Example output:

```
ubuntu@ubuntu:~/day0/q3$ ./energy
Calculating kinetic energy of a moving object:
Enter the object's mass in kilograms: 2
Enter the object's speed in meters per second: 3
The object has 6.00 joules of energy
```

Here is another output

```
ubuntu@ubuntu:~/day0/q3$ ./energy
Calculating kinetic energy of a moving object:
Enter the object's mass in kilograms: 24
Enter the object's speed in meters per second: 3.7
The object has 88.80 joules of energy
```

If you get an error when you run the *energy.c* program or you get a different output, you have made an editing error and you need to re-edit the file and fix the problem.

Task 5: Editing C Program (20 points)

Inside the *day0* directory, move to the *q4* directory. Inside this directory, create a file named *dna.c*. This program is used to compute the percentage of g-c and c-g combinations in a given DNA sequence. Place the following code into *gcCounter.c*:

```

#include<stdio.h>
#include<string.h>

int chCount(char *str, char x)
{
    int len, count=0, i;
    len = strlen(str);
    for (i=0;i<len;i++)
        if(str[i] == x) count++;
    return count;
}

float gcCount(char *seq)
{
    int lenSeq, cCount, gCount;
    float gcPercent;
    lenSeq = strlen(seq);
    gCount = chCount(seq, 'g');
    cCount = chCount(seq, 'c');
    gcPercent = 100 * (float)(gCount+cCount)/lenSeq;
    return gcPercent;
}

int main()
{
    char dna[100];
    printf("Enter a DNA sequence: ");
    scanf("%s",dna);
    printf("The g-c and c-g combinations in this dna is:%3.2f\n",gcCount(dna));
    return 0;
}

```

**Note: valid nucleotides found in DNA are denoted by the letters a, t, g, and c.*

Here are some example uses of the program:

```

ubuntu@ubuntu:~/day0/q4$ ./dna
Enter a DNA sequence: acaagatgccattgtcccccggcctcctgct
The g-c and c-g combinations in this dna is:62.50

```

Another example is:

```

ubuntu@ubuntu:~/day0/q4$ ./dna
Enter a DNA sequence: gattaca
The g-c and c-g combinations in this dna is:28.57

```

Task 6: Editing Source code (20 points)

Inside the *day0* directory, move to the *q5* directory. Download a source code (C Program) named *error.c* file from LMS. Move the *error.c* file to *q5* directory. The *error.c* is a broken code. There are few syntax errors and hence the code will not compile properly. Fix the program and make sure the program compiles with no warning or errors and provides the following output.

```

ubuntu@ubuntu:~/day0/q5$ ./error
Enter the degrees in fahrenheit ?100

```

```
Degrees Fahrenheit is: 100.00
Degrees Celsius is: 37.78
Degrees Kelvin is: 310.78
```

Another example is

```
ubuntu@ubuntu:~/day0/q5$ ./error
Enter the degrees in fahrenheit ?97.32
Degrees Fahrenheit is: 97.32
Degrees Celsius is: 36.29
Degrees Kelvin is: 309.29
```

Demonstrate your activities

Before demonstrating, make sure that you have all the files.

```
du -a
```

You should see the output:

```
20 ./q1/textlines.txt
24 ./q1
20 ./q2/manipulate.txt
24 ./q2
4 ./q3/energy.c
8 ./q3
4 ./q4/dna.c
8 ./q4
4 ./q5/error.c
8 ./q5
```

The order and numbers do not matter.