# Cosc 2P12Assignment 3

(Due date for assignment is Tuesday March 19th,16:00 est.)

**Part A**

In this part you are to create a tree using ascii characters. Prompt the user to enter an odd number and a character. For example, 5 and “G”. This will create a tree as follows.

 G
 GGG
 GGGGG
 G
 G

The widest part will be 5, the trunk will be a standard 2 units in length. Test your program using various numbers and characters.

**Part B**

A simple transposition cipher takes in a character and shifts it up/down by a fixed amount. For example, we just consider the lower case characters with a shift of 2, then an ‘a’ will become a ‘c’, a ‘b’ -> ‘d’; and a ‘z’ -> ‘b’, because we wrap around the alphabet. This is just a simple example of how wrapping can work.

The entire ASCII alphabet has 127 characters in it, this implies that any character can be shifted to a new location. Characters near the upper end with a shift of say 10 could wrap around to the bottom of the alphabet.

An ASCII character is nothing but a number, thus if we have the integer 65 we directly have the character ‘A’. Likewise we can add integers to a character to obtain a new character.

For this assignment you are to prompt the user to enter a string call it Str; followed by an integer N less than 127. Create the string Str\* where every character is shifted by N. Once you have done this print the encrypted string Str\* to the consol.

Using the Str\* and the knowledge of the shift N, reverse the process to obtain Str, print this to the consol.

E.g. Using imaginary value of N.

Input => “Attack at Dawn”
Encrypted =>”mwlo uu; ou;je”
Decrypted =>”Attack at Dawn”

To obtain the wrapping you can used X mod Y which gives the remainder of X/Y, MIPS has the **rem** pseudo instruction. E.g. X+N mod 127= ? where X is the character and N is the shift.

**Part C**

Prompt the user to enter a string, Call it Str.

Write a leaf function called Length(Str). Which returns an integer of the length of Str. This result is then stored into a memory variable Len.

Write the leaf function SubString(Str,B,E), which will return a pointer to a new string containing the characters from B to E-1. B cannot be less than 0, B must be less then E, If E > Len then the substring is from B to Str[Len-1]. If any usage errors occur the result is null, or 0.

Test your program by including the following test harness in your program. This is pseudo code!

Prompt for Str = “Over the Hill and Far Away”
Length(Str);
Print ( SubString(Str, -1, 5);
Print (SubString(Str, 0, 5);
Print (SubString(Str, 0, 30);
Print (SubString(Str, 10, 5);
Print (SubString(Str, 10, 17);

**Submission**

Submit through Brightspace. Include files with your source code for parts A,B and C

The End