**COMP.2030 HW 7**

For this assignment, which is a preparatory step to HW8 (binary bomb project), you need an account in the CS system. The CS account has the domain of cs.uml.edu, and looks like yourCSusername@cs.uml.edu (e.g. kim@cs.uml.edu).

Your UML account ??????@student.uml.edu would NOT work.

1. If you do not have a CS account, send an email to help@cs.uml.edu, indicating that you are in COMP.2030, or stop at DAN405 to sign up.
2. You can log into mercury server directly if you are on campus, but need to use a VPN if you access from home. The procedure for installing GlobalProtect is at <https://www.uml.edu/it/services/get-connected/remote-access/>.
3. Login to mercury by typing in Power Shell (or putty in Windows) or MAC Terminal

ssh -l yourCSusername **mercury.cs.uml.edu**

**Assignment a)**

1. Create a new file (emacs or vi) and enter the following function, and save it as **sum.c**.
2. Compile the file with ‘**gcc -S sum.c**’ command and check out if sum.s file is generated.
3. Copy the .s file into .txt file: ‘**cp sum.s sum.txt**’
4. In the sum.txt file, remove all directives and keep only x86 executable instructions.
5. In the sum.txt file, add comments after most x86 instructions in the same pseudo-C style as you did in MIPS programs.

int get\_sum(int X[], int n){

 int sum = 0;

 int i = 0;

 while (i < n){

 sum = sum + X[i];

 i++;

 }

 return sum;

}

**Assignment b)**

long switch\_eg

 (long x, long y, long z)

{

 long w = 1;

 switch(x) {

 case 1:

 w = y\*z;

 break;

 case 2:

 w = y/z;

 /\* Fall Through \*/

 case 3:

 w += z;

 break;

 case 5:

 case 6:

 w -= z;

 break;

 default:

 w = 2;

 }

 return w;

}

Repeat the five steps in Assignment a) above for the function switch\_eg on the right.

**What to submit:**

Merge two text files in Assignments a & b into a single file,

 and submit in BB.