Assignment One: Good Software Engg Practices

In this assignment, you will learn how to use tools to improve your productivity. To make the assignment meaningful , we will study Matrix Multiplication. You will first document how performance scales with matrix size. Subsequently, you will implement any version of Blocked Matrix Multiplication to improve the performance. A desirable goal is that your performance numbers improve by a factor of 5.

To measure time, you can use this microsecond timer code:

```
#include <sys/time.h>
#include <unistd.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>

const double micro = 1.0e-6;
double calculateTime()
{
   struct timeval TV;
   struct timezone TZ;

   int RC = gettimeofday(&TV,&TZ);

   if (RC == -1){
     printf("Bad call to gettimeofday\n");
     return(-1);
   }

   return ( ((double)TV.tv_sec ) + micro * ((double)  TV.tv_usec
));

}
```

Your matrixMulB.c should have the following structure and implement the functions with appropriate arguments:

```
void printMatrix(…){
  //Helper to print the Matrix passed as argument
}
void verify(…){
//Verifies that Blocked MatMul is correct
  }
int main(int argc, char **argv)
{
// Process command line arguments
// n, # repetitions, block size

// Start the timer
for (r=0; r<nreps; r++){ //For statistical correctness
   for (i=0; i<n; i++)
              for (j=0; j<n; j++){
                 sum = 0;
                 for (k=0; k<n; k++)
                           sum+= A[i][k] * B[k][j];
                 C[i][j] = sum;
              }
}
// Stop the timer
              calculateTime()
// Verify the results
              verify();
//The plotting can either be done in this place or you can save the times for different matrix sizes and gnuplot can be called externally.
}
```

What do you have to turn in:

- Document your work in a well-written report (pdf – no other extensions must be present in the Project directory) which discusses your findings carefully. Provide all data you plotted in tabular form. This should also have the following (obtained by executing the associated command). Place this in a file called env.txt in your Project directory.
  - OS name : uname -a
  - Your Machine : hostname
  - Compiler version number: gcc -v
- See if you can get the specs of the processor that you are using --- TLP, L1, L2/L3 cache
- gzip the Project directory and submit in Moodle. Please ensure that you run Make clean before you run gzip/tar -zxvf.