Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Compiler Laboratory: CS39003
3rd year CSE, 5th Semester

Assignment - 1: Annotating Assembly   *Marks: 50*
Assign Date: July 16, 2019   Submit Date: 23:55, July 22, 2019

1. Translate the following C program using GCC/Linux to the assembly language program of x86-64 (Intel 64-bit processor) without optimization.

```
cc -Wall -S asgn1.c
```

C Program: *asgn1.c*

```c
#include <stdio.h>

#define DIM 20

void ReadMat(int n, int data[][DIM]);
void TransMat(int n, int data[][DIM]);
int VectMult(int n, int firstMat[], int secondMat[]);
void MatMult(int n, int firstMat[][DIM], int secondMat[][DIM], int M[][DIM]);

int main()
{
        int n, i, j;
        int A[DIM][DIM],B[DIM][DIM],C[DIM][DIM];

        printf("Enter the dimension of a square matrix: ");
        scanf("%d", &n);
        printf("Enter the first matix (row-major): ");
        ReadMat(n, A);
        printf("Enter the second matix (row-major): ");
        ReadMat(n, B);
        MatMult(n, A, B, C);
        printf("\nThe result matrix:\n");
```

```c
        for(i=0; i<n; ++i){
                for(j=0; j<n; ++j) {
                        printf("%d ", C[i][j]);
                }
                printf("\n");
        }
        return 0;
}


void ReadMat(int n, int data[][DIM])
{
        int i, j;
        for(i=0; i<n; ++i) {
                for(j=0; j<n; ++j) {
                        scanf("%d", &data[i][j]);
                }
        }
}


void TransMat(int n, int data[][DIM])
{
        int i, j, t;
        for(i=0; i<n; i++) {
                for(j=0; j<i; j++) {
                        t = data[i][j];
                        data[i][j] = data[j][i];
                        data[j][i] = t;
                }
        }
        printf("\nThe transpose of the second matrix:\n");
        for(i=0; i<n; ++i){
                for(j=0; j<n; ++j) {
                        printf("%d ", data[i][j]);
                }
                printf("\n");
        }
}


int VectMult(int n, int firstMat[], int secondMat[])
{
        int i, result = 0;
        for(i=0; i<n; i++) {
                result+=firstMat[i]*secondMat[i];
        }
        return result;
}
```

2

```
void MatMult(int n, int firstMat[][DIM], int secondMat[][DIM], int M[][DIM])
{
        int i, j;
        TransMat(n, secondMat);
        for(i=0; i<n; i++) {
                for(j=0; j<n; j++) {
                        M[i][j] = VectMult(n, &firstMat[i][0], &secondMat[j][0]);
                }
        }
}
```

2. Rename the generated assembly file as *ass1_roll.s* (where *roll* is your roll number). Add comments for each of the assembly language instruction. Your comment should explain the functionality of the instruction and the connection to the original C program. Please make sure that your commented file can be compiled to generate executable file. Upload your file (**ass1_*roll*.s**) in Moodle server.

Note: *Comments without connection to C program will get partial marks.*