Computer Science & Engineering Department
I. I. T. Kharagpur

Compilers Laboratory: CS39003

3rd Year CSE: 5th Semester

Assignment - 1: Annotating Assembly — *Marks: 10*
Assign Date: *22nd July, 2015* — Submit Date: *23:55, 28th July, 2015*

1. Translate the following C program using GCC/Linux to the assembly language program of x86-64 (Intel 64-bit processor).

   `cc -Wall -S` <*file name*>`.c`

   Do not give any optimization option. The *file name* should be `ass1_`*roll*.c where *roll* is your roll number.

   Write comments in the assembly language code corresponding to the program <*file name*>`.s`. Comments should explain the corresponding assembly language instructions and also should clearly show the connection between the C program and the assembly language program.

```
/*
 * ass1.c Generate assembly code of x86-64 and comment
 */

#include <stdio.h>

#define SIZE 20

void ReadMatrix(int n, int data[][SIZE]);
void TransposeMatrix(int n, int data[][SIZE]);
int VectorMultiply(int n, int L[], int R[]);
void MultiplyMatrix(int n, int L[][SIZE], int R[][SIZE], int M[][SIZE]);

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

    printf("Enter the order: ");
    scanf("%d", &n);

    printf("Enter matix A in row-major: ");
    ReadMatrix(n, A);

    printf("Enter matix B in row-major: ");
    ReadMatrix(n, B);

    MultiplyMatrix(n, A, B, C);

    printf("The output matrix is:\n");
    for(i=0; i<n; ++i){
        for(j=0; j<n; ++j) printf("%d ", C[i][j]);
        putchar('\n');
    }

    return 0;
}
```

```
void ReadMatrix(int n, int data[][SIZE])
{
    int i, j ;

    for(i=0; i<n; ++i)
        for(j=0; j<n; ++j) scanf("%d", &data[i][j]);
    printf("The input matrix is:\n");
    for(i=0; i<n; ++i){
        for(j=0; j<n; ++j) printf("%d ", data[i][j]);
        putchar('\n');
    }
}

void TransposeMatrix(int n, int data[][SIZE])
{
    int i, j ;

    for(i=0; i<n; ++i)
        for(j=0; j<i; ++j)
        {
            int t = data[i][j];
            data[i][j] = data[j][i];
            data[j][i] = t;
        }
    printf("The transposed matrix is:\n");
    for(i=0; i<n; ++i){
        for(j=0; j<n; ++j) printf("%d ", data[i][j]);
        putchar('\n');
    }
}

int VectorMultiply(int n, int L[], int R[])
{
    int i, result = 0;

    for(i=0; i<n; ++i)
        result += L[i] * R[i];

    return result;
}

void MultiplyMatrix(int n, int L[][SIZE], int R[][SIZE], int M[][SIZE])
{
    int i, j;

    TransposeMatrix(n, R);

    for(i=0; i<n; ++i)
        for(j=0; j<n; ++j)
            M[i][j] = VectorMultiply(n, &L[i][0], &R[j][0]);
}
```

2. The commented assembly language program should remain syntactically correct.

3. Intel assembly language manual and other reading materials are available Moodle