CSE4001-PARALLEL AND DISTRIBUTED COMPUTING
lab-6

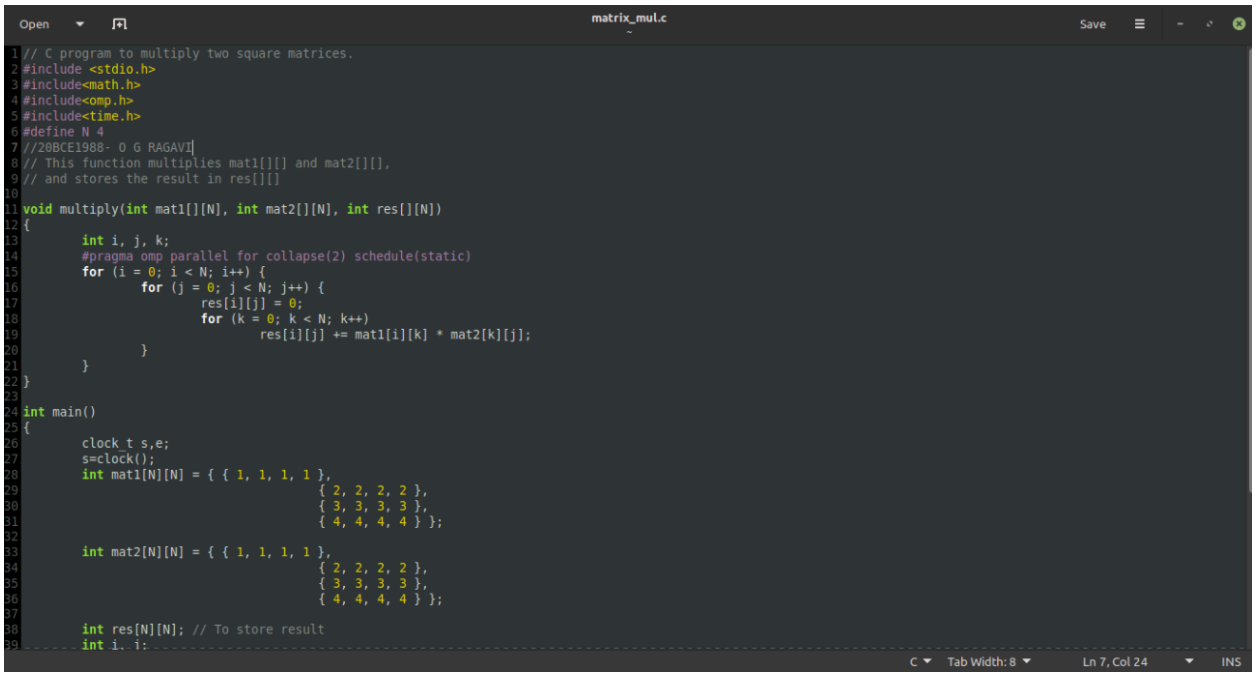
Name: O G Ragavi
Reg No: 20BCE1988
Lab: 6

1.OMP program to multiply 2 matrices:

Code:

```
// C program to multiply two square matrices.
#include <stdio.h>
#include<math.h>
#include<omp.h>
#include<time.h>
#define N 4

// This function multiplies mat1[][] and mat2[][],
// and stores the result in res[][]

void multiply(int mat1[][N], int mat2[][N], int res[][N])
{
    int i, j, k;
    #pragma omp parallel for collapse(2) schedule(static)
    for (i = 0; i < N; i++) {
        for (j = 0; j < N; j++) {
            res[i][j] = 0;
            for (k = 0; k < N; k++)
                res[i][j] += mat1[i][k] * mat2[k][j];
        }
    }
}

int main()
{
        clock_t s,e;
        s=clock();
    int mat1[N][N] = { { 1, 1, 1, 1 },
                    { 2, 2, 2, 2 },
                    { 3, 3, 3, 3 },
                    { 4, 4, 4, 4 } };

   int mat2[N][N] = { { 1, 1, 1, 1 },
                    { 2, 2, 2, 2 },
```

```
                { 3, 3, 3, 3 },
                { 4, 4, 4, 4 } };

   int res[N][N]; // To store result
    int i, j;
    multiply(mat1, mat2, res);

   printf("Result matrix is \n");
    for (i = 0; i < N; i++) {
        for (j = 0; j < N; j++)
            printf("%d ", res[i][j]);
        printf("\n");
    }
        e=clock();
        printf("Time taken %ld ",(e-s));
    return 0;
}
```



Output:

```
ex2@AB1210SCOPE70: ~
File  Edit  View  Search  Terminal  Help
ex2@AB1210SCOPE70:~$ gcc -o mul matrix_mul.c
ex2@AB1210SCOPE70:~$ ./mul
Result matrix is
10 10 10 10
20 20 20 20
30 30 30 30
40 40 40 40
ex2@AB1210SCOPE70:~$ gcc -o mul matrix_mul.c
ex2@AB1210SCOPE70:~$ ./mul
Result matrix is
10 10 10 10
20 20 20 20
30 30 30 30
40 40 40 40
ex2@AB1210SCOPE70:~$ gcc -o mul -fopenmp matrix_mul.c
ex2@AB1210SCOPE70:~$ ./mul
Result matrix is
9 10 10 10
20 20 20 20
30 30 30 30
40 60 40 40
ex2@AB1210SCOPE70:~$ gcc -o mul -fopenmp matrix_mul.c
ex2@AB1210SCOPE70:~$ ./mul
Result matrix is
10 10 10 10
20 20 20 20
30 30 30 30
20 40 40 40
Time taken 2726 ex2@AB1210SCOPE70:~$ gcc -o tsp tsp.c
gcc: error: tsp.c: No such file or directory
gcc: fatal error: no input files
compilation terminated.
ex2@AB1210SCOPE70:~$ gcc -o tsp tsp.c
ex2@AB1210SCOPE70:~$ ./tsp.c
bash: ./tsp.c: Permission denied
ex2@AB1210SCOPE70:~$ ./tsp
Enter Total Number of Cities:   4

Enter Cost Matrix
```

Graph:

2.OMP Program-Traveling salesman problem:

Code:

```
#include <stdio.h>
#include<omp.h>
#include<time.h>
#include<stdlib.h>
//20BCE1988- O G RAGAVI
int matrix[25][25], visited_cities[10], limit, cost = 0;

int tsp(int c)
{
int count, nearest_city = 999;
int minimum = 999, temp;
for(count = 0; count < limit; count++)
{
if((matrix[c][count] != 0) && (visited_cities[count] == 0))
{
if(matrix[c][count] < minimum)
{
minimum = matrix[count][0] + matrix[c][count];
```

```
}
temp = matrix[c][count];
nearest_city = count;
}
}
if(minimum != 999)
{
cost = cost + temp;
}
return nearest_city;
}

void minimum_cost(int city)
{
int nearest_city;
visited_cities[city] = 1;
printf("%d ", city + 1);
nearest_city = tsp(city);
if(nearest_city == 999)
{
nearest_city = 0;
printf("%d", nearest_city + 1);
cost = cost + matrix[city][nearest_city];
return;
}
minimum_cost(nearest_city);
}

int main()
{
int i, j;
clock_t s,e;
s=clock();
printf("Enter Total Number of Cities:\t");
scanf("%d", &limit);
printf("\nEnter Cost Matrix\n");
```

```
for(i = 0; i < limit; i++)
{
printf("\nEnter %d Elements in Row[%d]\n", limit, i + 1);
for(j = 0; j < limit; j++)
{
scanf("%d", &matrix[i][j]);
}
visited_cities[i] = 0;
}

printf("\nEntered Cost Matrix\n");
  #pragma omp parallel for
for(i = 0; i < limit; i++)
{
   printf("\n");
   for(j = 0; j < limit; j++)
   {
     printf("%d ", matrix[i][j]);
   }
}
printf("\n\nPath:\t");
minimum_cost(0);
printf("\n\nMinimum Cost: \t");
printf("%d\n", cost);
e=clock();
printf("\nTime taken is %ld ",(e-s));
return 0;
}
```

Output:

```
ex2@AB1210SCOPE70: ~
File  Edit  View  Search  Terminal  Help
ex2@AB1210SCOPE70:~$ gcc -o tsp -fopenmp tsp.c
tsp.c: In function ‘main’:
tsp.c:81:2: warning: implicit declaration of function ‘pritf’; did you mean ‘printf’? [-Wimplicit-function-declaration]
   81 |  pritf("\nTime taken is %ld ",(e-s));
      |  ^~~~~
      |  printf
/usr/bin/ld: /tmp/ccPUTX0W.o: in function `main':
tsp.c:(.text+0x40f): undefined reference to `pritf'
collect2: error: ld returned 1 exit status
ex2@AB1210SCOPE70:~$ gcc -o tsp -fopenmp tsp.c
ex2@AB1210SCOPE70:~$ ./tsp
Enter Total Number of Cities:   4

Enter Cost Matrix

Enter 4 Elements in Row[1]
1 2 3 4

Enter 4 Elements in Row[2]
5 6 7 8

Enter 4 Elements in Row[3]
3 4 5 6

Enter 4 Elements in Row[4]
9 8 4 3

Entered Cost Matrix

3 4 5 6

1 2 3 4
9 8 4 3 5 6 7 8

Path:   1 4 3 2 1

Minimum Cost:   17

ex2@AB1210SCOPE70:~$
```

Graph: