Sagib Javed
Roll # 2222

a-
The nested loops inside the add() function should be parallelized because they are performing the same operation on different elements of the matrices.

b-
The loops in the add() function have no loop-carried dependencies since each element of the output matrix "C" is computed independently using only the corresponding elements of the input matrices "A" and "B".

c-
Here's the parallel version of the program using OpenMP:

```
....
#include <stdio.h>
#include< omp.h>
# define N 16
Void add( int A[][N], int B[][N], int C[][N])
  {
    int i,j;
# program omp parallel for private(i,j)
    for (i=0;i<N;i++)
  {
    for (j=0;j<N;j++)
  {
    C[i][j] = A[i][j] + B[i][j];
  }
}
```

STAR

M T W T F S S

Date:

```
}
int main( )
{
    int C[N][N];
  int A[N][N] = {(1,1,1,1....),(2,2,2,2.....),
        (3,3,3,3....), ------(4,4,4,4....)};

add(A,B,C);
Printf("Result Matrix is \n");
   int i,j;
    for(i=0; i<N; i++)
      {
        for(j=0; j<N;j++)
    {
     Printf("%d", C[i][j]);
   }
     Printf("\n");
}
     return 0;
}.
```