Implement any scheme to find the optimal solution for the Traveling Sales Person problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation

```
#include<stdio.h>


int s, ver, c[100][100];

float optimum = 999, sum;


/* function to swap array elements */

void swap(int v[], int i, int j) {

   int t;

   t = v[i];

   v[i] = v[j];

   v[j] = t;

}


/* recursive function to generate permutations */

void brute_force(int v[], int n, int i) {

   int j, sum1, k;


   if (i == n) {

      if (v[0] == s) {

         for (j = 0; j < n; j++)

            printf("%d ", v[j]);


         sum1 = 0;

         for (k = 0; k < n - 1; k++) {

            sum1 = sum1 + c[v[k]][v[k + 1]];
            }
            sum1 = sum1 + c[v[n - 1]][s];

            printf("sum = %d\n", sum1);

            if (sum1 < optimum)
                optimum = sum1;
        }
    } else {
        for (j = i; j < n; j++) {
            swap(v, i, j);
            brute_force(v, n, i + 1);
            swap(v, i, j);
        }
    }
}

void nearest_neighbour() {
    int min, p, i, j, vis[20], from;

    for (i = 1; i <= ver; i++)
        vis[i] = 0;

    vis[s] = 1;
    from = s;
    sum = 0;

    for (j = 1; j < ver; j++) {
        min = 999;
        for (i = 1; i <= ver; i++) {
            if (vis[i] != 1 && c[from][i] < min && c[from][i] != 0) {
                min = c[from][i];
                p = i;
            }
        }

        vis[p] = 1;
        from = p;
        sum = sum + min;
    }

    sum = sum + c[from][s];
}

int main() {
    int v[100], i, j;

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

    for (i = 0; i < ver; i++)
        v[i] = i + 1;

    printf("Enter cost matrix\n");
    for (i = 1; i <= ver; i++)
        for (j = 1; j <= ver; j++)
            scanf("%d", &c[i][j]);
    printf("Enter source: ");
    scanf("%d", &s);

    brute_force(v, ver, 0);

    printf("\nOptimum solution with brute force technique is = %f\n", optimum);

    nearest_neighbour();

    printf("\nSolution with nearest neighbour technique is = %f\n", sum);
    printf("The approximation value is = %f%%\n", ((sum / optimum) - 1) * 100);

    return 0;
}
```