35.

```
#include <stdio.h>
int parent[10];
int find(int i) {
   while (parent[i]) i = parent[i];
   return i;
}
int uni(int i, int j) {
   if (i != j) {
      parent[j] = i;
      return 1;
   }
   return 0;
}
int main() {
   int n, ne = 1, min, a, b, u, v, i, j;
   int cost[10][10], mincost = 0;
   printf("Enter number of vertices: ");
   scanf("%d", &n);
   printf("Enter cost adjacency matrix:\n");
   for (i = 0; i < n; i++)
      for (j = 0; j < n; j++) {
         scanf("%d", &cost[i][j]);
         if (cost[i][j] == 0) cost[i][j] = 999;
      }
   printf("Edges of MST:\n");
   while (ne < n) {
      min = 999;
      for (i = 0; i < n; i++)
         for (j = 0; j < n; j++)
```

```
Enter number of vertices: 3
Enter cost adjacency matrix:
5 4 7
8 7 6
3 2 8
Edges of MST:
2 - 1 : 2
2 - 0 : 3
Minimum cost = 5
```

```
            if (cost[i][j] < min) {
                min = cost[i][j];
                a = u = i;
                b = v = j;
            }
        u = find(u);
        v = find(v);
        if (uni(u, v)) {
            printf("%d - %d : %d\n", a, b, min);
            mincost += min;
            ne++;
        }
        cost[a][b] = cost[b][a] = 999;
    }
    printf("Minimum cost = %d\n", mincost);
    return 0;
}
```