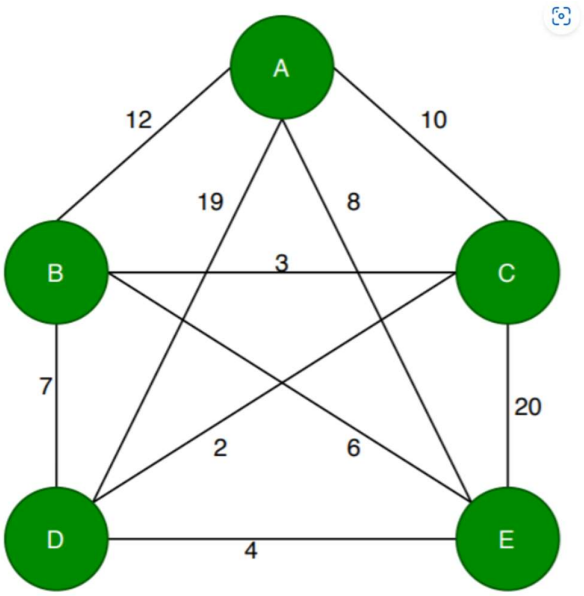
.

Department of Computer Science and Engineering (Data Science)
Experiment 6
(Dynamic Programming)

Name – DIVYESH KHUNT **SAP ID:60009210116**

Aim: Implementation of travelling salesperson using dynamic programming.

THEORY:



The goal is to find the shortest possible path for the tour that starts from the origin city, traverses the graph while only visiting the other cities or nodes once, and returns to the origin city.

Time complexity:

In the dynamic algorithm for TSP, the number of possible subsets can be at most $N \times 2^N$. Each subset can be solved in $\mathcal{O}(N)$ times. **Therefore, the time complexity of this algorithm would be** $\mathcal{O}(\mathbf{N^2 \times 2^N})$.

.

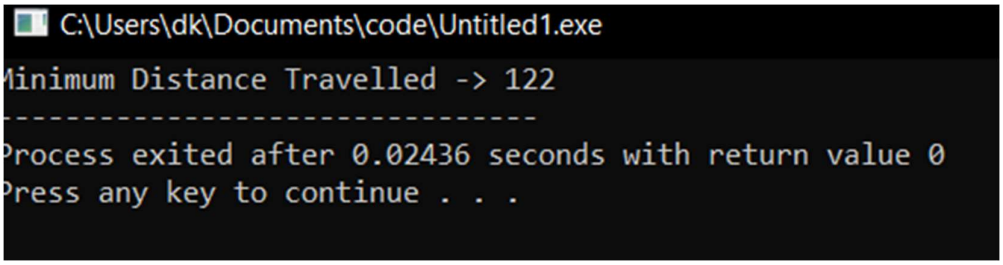
\

CODE:

```
#include<iostream>
using namespace std;
#define MAX 9999
int n=4;
int distan[20][20] = {{0, 22, 26, 30},
   {30, 0, 45, 35},
   {25, 45, 0, 60},
   {30, 35, 40, 0}
};
int completed_visit = (1<<n) -1;
int DP[32][8];
int TSP(int mark, int position){
   if(mark==completed_visit) {
      return distan[position][0];
   }
   if(DP[mark][position]!=-1) {
      return DP[mark][position];
   }
   int answer = MAX;
   for(int city=0; city<n; city++) {
      if((mark&(1<<city))==0) {
         int newAnswer = distan[position][city] + TSP( mark|(1<<city),city);
         answer = min(answer, newAnswer);
      }
   }
   return DP[mark][position] = answer;
}
int main(){
   for(int i=0; i<(1<<n); i++) {
      for(int j=0; j<n; j++) {
         DP[i][j] = -1;
      }
   }
   cout << "Minimum Distance Travelled -> " << TSP(1,0);
   return 0;
}
```

.

OUTPUT:


C:\Users\dk\Documents\code\Untitled1.exe
Minimum Distance Travelled -> 122

Process exited after 0.02436 seconds with return value 0
Press any key to continue . . .