Traveling Salesperson

Competitive Programming

Manager	Keeps team on track	
Recorder	Records decisions	
Reporter	Reports to class	
Reflector	Assesses team performance	

Part 1

Code

Here is the code for TSP.

```
vvi dp(16,vi(65536));
int n;
int tsp(vvi &costs, int &mx, int cur, int state) {
   if (dp[cur][state]>0) return dp[cur][state];
   if (state == mx) return costs[cur][0]; // return home

   int minleft = INF;
   int bit=2;

   for(int i=1; i<n; ++i)  {
     if ( (state & bit) == 0) { // i not visited
        minleft = min(minleft, costs[cur][i] + tsp(costs,mx,i,state | bit));
     }
     bit <<= 1;
   }
   return dp[cur][state]=minleft;
}

int main() {
   cin >> n;
   vvi adj(n);

   for(i=0; i<n; ++i)
      for(j=0; j<n; ++j) {
         cin >> c;
         adj[i].push_back(c);
      }

   mx = (1 << n) - 1;
   cout << "Best path has cost " << tsp(adj,mx,0,1) << endl;
}
```

Questions

Application

Costs:

	0	1	2
0	0	8	6
1	7	0	5
2	3	9	0

DP:

	111	110	101	100	011	010	001
0							
1							
2							