**package** lp3;

**import** java.util.Arrays;

**public** **class** TSPNearestNeighbor {

**public** **static** **void** main(String[] args) {

**int**[][] graph = {

{0, 29, 20, 21},

{29, 0, 15, 12},

{20, 15, 0, 25},

{21, 12, 25, 0}

};

**int**[] tour = *solveTSP*(graph);

System.***out***.println("Approximate tour: " + Arrays.*toString*(tour));

}

**public** **static** **int**[] solveTSP(**int**[][] graph) {

**int** n = graph.length;

**int**[] tour = **new** **int**[n];

**boolean**[] visited = **new** **boolean**[n];

tour[0] = 0;

visited[0] = **true**;

**for** (**int** i = 1; i < n; i++) {

**int** nearestCity = -1;

**int** minDistance = Integer.***MAX\_VALUE***;

**for** (**int** j = 0; j < n; j++) {

**if** (!visited[j] && graph[tour[i - 1]][j] < minDistance) {

nearestCity = j;

minDistance = graph[tour[i - 1]][j];

}

}

tour[i] = nearestCity;

visited[nearestCity] = **true**;

}

**return** tour;

}

}