**Name:Shraddha Zoman**

**Roll no:TYCOD298**

**Problem Statement**

Let there be N students and N clubs. Any student can be assigned to any club, incurring some cost that may vary depending on the student club assignment. It is required to allocate all clubs by assigning exactly one student to each club and exactly one club to each agent in such a way that the total cost of the assignment is minimized. Implement club assignment problem using Branch and bound.

Code:

import java.util.\*;

class C implements Comparable<C> {

int sl;

int[] ac;

int c;

int lb;

public C(int sl, int[] ac, int c, int lb) {

this.sl = sl;

this.ac = ac.clone();

this.c = c;

this.lb = lb;

}

@Override

public int compareTo(C o) {

return Integer.compare(this.lb, o.lb);

}

}

public class A6 {

public static int aClubs(int[][] cm, int[] res) {

int n = cm.length;

PriorityQueue<C> pq = new PriorityQueue<>();

int[] ia = new int[n];

Arrays.fill(ia, -1);

C r = new C(0, ia, 0, lb(cm, ia, 0));

pq.add(r);

while (!pq.isEmpty()) {

C cur = pq.poll();

if (cur.sl == n) {

System.arraycopy(cur.ac, 0, res, 0, n);

return cur.c;

}

int st = cur.sl;

for (int cl = 0; cl < n; cl++) {

if (!clubAsg(cur.ac, cl)) {

int[] newAc = cur.ac.clone();

newAc[st] = cl;

int newC = cur.c + cm[st][cl];

int newLb = lb(cm, newAc, st + 1);

C newNode = new C(st + 1, newAc, newC, newLb);

pq.add(newNode);

}

}

}

return -1;

}

private static boolean clubAsg(int[] ac, int cl) {

for (int a : ac) {

if (a == cl) return true;

}

return false;

}

private static int lb(int[][] cm, int[] ac, int sl) {

int n = cm.length;

int l = 0;

for (int i = 0; i < sl; i++) {

l += cm[i][ac[i]];

}

for (int i = sl; i < n; i++) {

int min = Integer.MAX\_VALUE;

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

if (!clubAsg(ac, j)) {

min = Math.min(min, cm[i][j]);

}

}

l += min;

}

return l;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter number of students/clubs: ");

int n = sc.nextInt();

int[][] cm = new int[n][n];

System.out.println("Enter the cost matrix: ");

for (int i = 0; i < n; i++) {

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

cm[i][j] = sc.nextInt();

}

}

int[] res = new int[n];

int minC = aClubs(cm, res);

System.out.println("The minimum cost of assigning clubs is: " + minC);

System.out.println("Club assignments (student -> club): ");

for (int i = 0; i < n; i++) {

System.out.println("Student " + (i + 1) + " -> Club " + (res[i] + 1));

}

}

}

Output:

PS D:\Java> java assign6.java

Enter number of students/clubs: 4

Enter the cost matrix:

4

4

3

2

2

1

2

6

7

5

9

4

3

2

2

5

The minimum cost of assigning clubs is: 11

Club assignments (student -> club):

Student 1 -> Club 3

Student 2 -> Club 1

Student 3 -> Club 4

Student 4 -> Club 2