```
int P = 2;
int L_count = 3;
int K_count = 4;

range Locations = 1..L_count;
range ReferencePoints = 1..K_count;

float L[Locations][1..2] = [[2,6],[1,2],[3,3]];
float K[ReferencePoints][1..2] = [[1,3],[2,3],[2,5.3],[3.4,3]];

dvar boolean x[Locations];
dvar float+ total_distance;

minimize total_distance;

subject to {

    sum(i in Locations) x[i] == P;
    total_distance == sum(i in Locations) (x[i] * min(j in ReferencePoints) pow(pow(L[i][1] - K[j][1], 2) +
pow(L[i][2] - K[j][2], 2), 0.5));
}

execute DISPLAY {
    writeln("Minimalna suma odległości: ", total_distance);
    write("Wybrane lokacje: ");
    for (i in Locations) {
        write(x[i], " ");
    }
}
```