Week 6 Homework

Submit your homework as a MS Word or PDF document this week. Solve the 4 problems below. In each case cut and paste the model and results into this document.

1. Problem 1 (7 pts) – use CP in OPL to solve textbook problem 12.9-2. This is very similar to one of the examples in the presentation this week. Paste your model and results here:

using CP;

//range i = 1..5;

dvar int x[1..5];

maximize 5\*x[1] - x[1]^2 + 8\*x[2] - x[2]^2 + 10\*x[3]

- x[3]^2 + 15\*x[4] - x[4]^2 + 20\*x[5] - x[5]^2;

subject to {

ct1:

allDifferent(x);

ct2:

x[1] + x[3] + x[4] <= 25;

ct3:

x[1] in {3, 6, 12};

ct4:

x[2] in {3, 6};

ct5:

x[3] in {3, 6, 9, 12};

ct6:

x[4] in {6, 12};

ct7:

x[5] in {9, 12, 15, 18};

}

// solution with objective 138

x = [3 6 9 12 15];

1. Problem 2 (8 pts) – use CP in OPL to solve textbook problem 12.9-5. You’ll need to create a “dummy” race as a place to assign the fifth swimmer. Paste your model and results here:

using CP;

//

//range R = 1..5;

//dvar int x[R] in 1..5;

//range Q = 1..5;

//float z1[Q] = [37.7, 43.4, 33.3, 29.2, 0];

//float z2[Q] = [32.9, 33.1, 28.5, 26.4, 0];

//float z3[Q] = [33.8, 42.2, 38.9, 29.6, 0];

//float z4[Q] = [37.0, 34.7, 30.4, 28.5, 0];

//float z5[Q] = [35.4, 41.8, 33.6, 31.1, 0];

//

//minimize

//

// z1[x[1]] +

// z2[x[2]] +

// z3[x[3]] +

// z4[x[4]] +

// z5[x[5]];

//

//subject to {

// allDifferent(x);

//}

range R = 1..5;

dvar int x[R] in R;

float z[R][R] = [[37.7, 43.4, 33.3, 29.2, 0],

[32.9, 33.1, 28.5, 26.4, 0],

[33.8, 42.2, 38.9, 29.6, 0],

[37.0, 34.7, 30.4, 28.5, 0],

[35.4, 41.8, 33.6, 31.1, 0]];

minimize

sum(r in R) z[r][x[r]];

subject to {

allDifferent(x);

}

// solution with objective 126.2

x = [4 3 1 2 5];

1. Problem 3 (7 pts) – solve the Traveling Salesman Problem using CP as discussed in the presentation and in problem 12-9.8. Write a CP program in OPL to solve the TSP introduced on page 621 of your book. You can start with the file tsp\_cp\_skeleton.mod in the download packet. The model file includes the cost matrix associated with the graph in figure 14.4. Note that we are using a very large value of $M$ as the cost between cities which are not connected to prevent the solution from using those connections. Paste your model and results here:

using CP;

int n = 7;

int M = 1000;

int c[1..n][1..n] =

[ [ 0, 12, 10, M, M, M, 12],

[12, 0, 8, 12, M, M, M],

[10, 8, 0, 11, 3, M, 9],

[ M, 12, 11, 0, 11, 10, M],

[ M, M, 3, 11, 0, 6, 7],

[ M, M, M, 10, 6, 0, 9],

[12, M, 9, M, 7, 9, 0]];

range R = 2..n;

dvar int x[R] in R;

minimize

// c[1][x[2]] +

// c[x[2]][x[3]] +

// c[x[3]][x[4]] +

// c[x[4]][x[5]] +

// c[x[5]][x[6]] +

// c[x[6]][x[7]] +

// c[x[7]][1];

c[1][x[2]] +

sum(r in 2..n-1) c[x[r]][x[r + 1]] +

c[x[n]][1];

subject to {

allDifferent(x);

}

// solution with objective 63

x = [3 5 7 6 4 2];

Problem 4 (8 pts) – use CP in OPL to solve the Reliable Construction Company construction scheduling problem described in the supplemental textbook section 22.1 (in download packet). You’ll want to study the example sched\_intro.mod that is included with OPL (File -> New -> Example …) which is also in the download packet for convenience (we deleted some stuff in the version in the download packet for simplicity). Paste your model and results here:

using CP;

dvar interval A size 2;

dvar interval B size 4;

dvar interval C size 10;

dvar interval D size 6;

dvar interval E size 4;

dvar interval F size 5;

dvar interval G size 7;

dvar interval H size 9;

dvar interval I size 7;

dvar interval J size 8;

dvar interval K size 4;

dvar interval L size 5;

dvar interval M size 2;

dvar interval N size 6;

//minimize startOf(A) + startOf(B) + startOf(C) + startOf(D) + startOf(E) + startOf(F) + startOf(G) + startOf(H) + startOf(I) + startOf(J) + startOf(K) + startOf(L) + startOf(M) + startOf(N);

subject to {

endBeforeStart(A, B);

endBeforeStart(B, C);

endBeforeStart(C, D);

endBeforeStart(C, E);

endBeforeStart(E, F);

endBeforeStart(D, G);

endBeforeStart(E, H);

endBeforeStart(G, H);

endBeforeStart(C, I);

endBeforeStart(F, J);

endBeforeStart(I, J);

endBeforeStart(J, K);

endBeforeStart(J, L);

endBeforeStart(H, M);

endBeforeStart(K, N);

endBeforeStart(L, N);

}

execute {

writeln("A : " + A.start + ".." + A.end);

writeln("B : " + B.start + ".." + B.end);

writeln("C : " + C.start + ".." + C.end);

writeln("D : " + D.start + ".." + D.end);

writeln("E : " + E.start + ".." + E.end);

writeln("F : " + F.start + ".." + F.end);

writeln("G : " + G.start + ".." + G.end);

writeln("H : " + H.start + ".." + H.end);

writeln("I : " + I.start + ".." + I.end);

writeln("J : " + J.start + ".." + J.end);

writeln("K : " + K.start + ".." + K.end);

writeln("L : " + L.start + ".." + L.end);

writeln("M : " + M.start + ".." + M.end);

writeln("N : " + N.start + ".." + N.end);

}

// solution

A : 0..2

B : 2..6

C : 6..16

D : 16..22

E : 16..20

F : 20..25

G : 22..29

H : 29..38

I : 16..23

J : 25..33

K : 33..37

L : 33..38

M : 38..40

N : 38..44