**CSE 579**

**Module 5 Graded Assignment**

**Template for clingo Work**

Problem 1

|  |  |
| --- | --- |
| Input  Program | %blocks.lp  % every block is a location  location(B) :- block(B).  % the table is a location  location(table).  % two blocks can't be on the same block at the same time  :- 2{on(BB,B,T)}, block(B), T = 0..m.  % effect of moving a block  on(B,L,T+1) :- move(B,L,T).  % concurrent actions are limited by num of grippers  :- not {move(BB,LL,T)} grippers, T = 0..m-1.  % a block can be moved only when it is clear  :- move(B,L,T), on(B1,B,T).  % a block can't be moved onto a block that is being moved also  :- move(B,B1,T), move(B1,L,T).  % fluents are initially exogenous  1{on(B,LL,0):location(LL)}1 :- block(B).  % uniqueness and existence of value constraints  :- not 1{on(B,LL,T)}1, block(B), T=1..m.  % actions are exogenous  {move(B,L,T)} :- block(B), location(L), T = 0..m-1.  % commonsense law of inertia  {on(B,L,T+1)} :- on(B,L,T), T < m.  % Answer  :- {on(B,table,T)} > l, T = 0..m.  #show move/3.  %===========================================  % Blocks-scenario.lp  %===========================================  block(1..6).  % initial state  :- not on(1,2,0; 2,table,0; 3,4,0; 4,table,0; 5,6,0; 6,table,0).  % goal  :- not on(3,2,m; 2,1,m; 1,table,m; 6,5,m; 5,4,m; 4,table,m). |
| Command  Lines | clingo blocks.lp blocks-scenario.lp -c l=3 -c grippers=2 -c m=5  clingo blocks.lp blocks-scenario.lp -c l=4 -c grippers=2 -c m=4 |
| Outputs  of clingo |  |
| Answer  to Questions | |  |  | | --- | --- | | l | Number of steps(m) | | 3 | 5 | | 4 | 4 | |

Problem 2

|  |  |
| --- | --- |
| Input  Program | %Blocks.lp  % every block is a location  location(B) :- block(B).  % the table is a location  location(table).  % two blocks can't be on the same block at the same time  :- 2{on(BB,B,T)}, block(B), T = 0..m.  % effect of moving a block  on(B,L,T+1) :- move(B,L,T).  % concurrent actions are limited by num of grippers  :- not {move(BB,LL,T)} grippers, T = 0..m-1.  % a block can be moved only when it is clear  :- move(B,L,T), on(B1,B,T).  % a block can't be moved onto a block that is being moved also  :- move(B,B1,T), move(B1,L,T).  % fluents are initially exogenous  1{on(B,LL,0):location(LL)}1 :- block(B).  % uniqueness and existence of value constraints  :- not 1{on(B,LL,T)}1, block(B), T=1..m.  % actions are exogenous  {move(B,L,T)} :- block(B), location(L), T = 0..m-1.  % commonsense law of inertia  {on(B,L,T+1)} :- on(B,L,T), T < m.  % Answer  under(B1,B2,T) :- on(B2,B1,T).  :- on(B1,B2,T), under(B1,B2,T).  under(B1,B3,T) :- under(B1,B2,T), under(B2,B3,T).  :- on(B1,B1,T).  block(1..6).  #show move/3. |
| Command  Line | clingo blocks.lp -c grippers=2 -c m=0 0 |
| Output  of clingo | .  .  .  . |
| Answer  to Questions | How many valid states are there when there are 6 blocks? (Note that the limitation of blocks introduced in question 1 is not considered here.)  4051 |

Problem 3

|  |  |
| --- | --- |
| Input  Program | % Blocks.lp  % every block is a location  location(B) :- block(B).  % the table is a location  location(table).  % two blocks can't be on the same block at the same time  :- 2{on(BB,B,T)}, block(B), T = 0..s.  % effect of moving a block  on(B,L,T+1) :- move(B,L,T).  % concurrent actions are limited by num of grippers  :- not {move(BB,LL,T)} grippers, T = 0..s-1.  % a block can be moved only when it is clear  :- move(B,L,T), on(B1,B,T).  % a block can't be moved onto a block that is being moved also  :- move(B,B1,T), move(B1,L,T).  % fluents are initially exogenous  1{on(B,LL,0):location(LL)}1 :- block(B).  % uniqueness and existence of value constraints  :- not 1{on(B,LL,T)}1, block(B), T=1..s.  % actions are exogenous  {move(B,L,T)} :- block(B), location(L), T = 0..s-1.  % commonsense law of inertia  {on(B,L,T+1)} :- on(B,L,T), T < s.  % serializing the actions  :- move(B1,L1,T), on(B2,L1,T), move(B2,L2,T), block(B2), T = 1..s-1.  #show move/3.  %===========================================  % Blocks-scenario.lp  %===========================================  block(a;b;c;d;e;f;g;h;i;j;k;l;m;n;o).  % initial state  :- not on(m,table,0; l,m,0; a,l,0; b,a,0; c,b,0; o,table,0; n,o,0; d,n,0; e,d,0; j,e,0; k,j,0; f,table,0; g,f,0; h,g,0; i,h,0).  % maxstep  :- not on(e,j,s; a,e,s; n,a,s; i,d,s; h,i,s; m,h,s; o,m,s; k,g,s; c,k,s; b,c,s; l,b,s). |
| Command  Line | Please only show the command line that outputs the minimal length plan.  clingo blocks.lp blocks-scenario.lp -c grippers=2 -c s=9 |
| Output  of clingo |  |

Problem 4

|  |  |
| --- | --- |
| Input  Program | %Blocks.lp  % every block is a location  location(B) :- block(B).  % the table is a location  location(table).  % two blocks can't be on the same block at the same time  :- 2{on(BB,B,T)}, block(B), T = 0..s.  % effect of moving a block  on(B,L,T+1) :- move(B,L,T).  % concurrent actions are limited by num of grippers  :- not {move(BB,LL,T)} grippers, T = 0..s-1.  % a block can be moved only when it is clear  :- move(B,L,T), on(B1,B,T).  % a block can't be moved onto a block that is being moved also  :- move(B,B1,T), move(B1,L,T).  % fluents are initially exogenous  1{on(B,LL,0):location(LL)}1 :- block(B).  % uniqueness and existence of value constraints  :- not 1{on(B,LL,T)}1, block(B), T=1..s.  % actions are exogenous  {move(B,L,T)} :- block(B), location(L), T = 0..s-1.  % commonsense law of inertia  {on(B,L,T+1)} :- on(B,L,T), T < s.  % serializing the actions  :- move(B1,L1,T), on(B2,L1,T), move(B2,L2,T), block(B2), T = 1..s-1.  % minimize number of actions (moves)  #minimize{1,B,L,T:move(B,L,T)}.  #show move/3.  %===========================================  % Blocks-scenario.lp  %===========================================  block(a;b;c;d;e;f;g;h;i;j;k;l;m;n;o).  % initial state  :- not on(m,table,0; l,m,0; a,l,0; b,a,0; c,b,0; o,table,0; n,o,0; d,n,0; e,d,0; j,e,0; k,j,0; f,table,0; g,f,0; h,g,0; i,h,0).  % maxstep  :- not on(e,j,s; a,e,s; n,a,s; i,d,s; h,i,s; m,h,s; o,m,s; k,g,s; c,k,s; b,c,s; l,b,s). |
| Command  Line | You should write multiple command lines below.  clingo blocks.lp blocks-scenario.lp -c grippers=2 -c s=8  clingo blocks.lp blocks-scenario.lp -c grippers=2 -c s=9  clingo blocks.lp blocks-scenario.lp -c grippers=2 -c s=10 -t3 |
| Output  of clingo | You should write multiple outputs, one for each command. These outputs serve as the evidences of your answer to the question below.  **CASE s=8: It is Unsatisfiable for S=8 with grippers=2**    **CASE s=9:**    **CASE s=10:** |
| Answer  to Questions | What is the least number of actions when maxstep m is 8, 9, and 10?  For 10 when I tried without -t3, It was not coming for 10 mins, so I killed it and used -t3   |  |  | | --- | --- | | m | least number of actions | | 8 | Unsatisfiable | | 9 | 16 | | 10 | 15 | |