**CSE 579**

**Programming Assignment 1**

**Template for clingo Work**

Problem 1

|  |  |
| --- | --- |
| Input  Program | % Define the size of the chessboard  size(8).  % Generate positions for queens ensuring they are not in the middle 4x4 square  { queen(X,Y) : X = 1..8, Y = 1..8, not middle(X,Y) } = 8.  % Define the middle 4x4 squares  middle(3..6,3..6).  % Constraint: No two queens on the same row  :- queen(X,Y1), queen(X,Y2), Y1 != Y2.  % Constraint: No two queens on the same column  :- queen(X1,Y), queen(X2,Y), X1 != X2.  % Constraint: No two queens on the same diagonal  :- queen(X1,Y1), queen(X2,Y2), X1 != X2, abs(X1-X2) = abs(Y1-Y2).  % Display the solution.  #show queen/2. |
| Command  Line | clingo ./program\_1.txt 0 |
| Output  of clingo (Truncated) | queen(2,3) queen(5,7) queen(1,4) queen(4,8) queen(7,6) queen(3,1) queen(8,5) queen(6,2)  Answer: 16  queen(5,7) queen(2,5) queen(4,8) queen(1,6) queen(3,1) queen(7,4) queen(6,2) queen(8,3)  Answer: 17  queen(2,3) queen(6,8) queen(4,7) queen(1,5) queen(3,2) queen(8,6) queen(7,4) queen(5,1)  Answer: 18  queen(6,7) queen(1,3) queen(2,5) queen(4,8) queen(3,2) queen(8,6) queen(7,4) queen(5,1)  Answer: 19  queen(6,7) queen(2,5) queen(4,8) queen(1,6) queen(3,2) queen(7,4) queen(5,1) queen(8,3)  Answer: 20  queen(6,7) queen(2,4) queen(5,8) queen(1,5) queen(3,2) queen(8,6) queen(4,1) queen(7,3)  Answer: 21  queen(2,4) queen(4,7) queen(1,5) queen(3,8) queen(8,6) queen(5,2) queen(7,3) queen(6,1)  Answer: 22  queen(2,3) queen(6,8) queen(4,7) queen(1,5) queen(7,6) queen(3,1) queen(5,2) queen(8,4)  Answer: 23  queen(2,3) queen(4,7) queen(1,5) queen(3,8) queen(7,6) queen(5,2) queen(8,4) queen(6,1)  Answer: 24  queen(6,7) queen(1,3) queen(2,5) queen(4,8) queen(7,6) queen(3,1) queen(5,2) queen(8,4)  SATISFIABLE  Models : 24  Calls : 1  Time : 0.061s (Solving: 0.06s 1st Model: 0.00s Unsat: 0.03s)  CPU Time : 0.061s |

Problem 2

|  |  |
| --- | --- |
| Input  Program | #const n = 10.  % Define the cells of the chessboard in which the Queens can be placed.  {queen\_cell(R, 1..n)} = 1 :- R = 1..n.  % 2 queens not in the similar column  :- queen\_cell(R1, C), queen\_cell(R2, C), R1 != R2.  % queens not in diagonal  :- queen\_cell(R1, C1), queen\_cell(R2, C2), R1 != R2, |R1 - R2| = |C1 - C2|. |
| Command  Line | clingo -c n=3 p2.txt 0  clingo -c n=4 p2.txt 0  clingo -c n=5 p2.txt 0  clingo -c n=6 p2.txt 0  clingo -c n=7 p2.txt 0  clingo -c n=8 p2.txt 0  clingo -c n=9 p2.txt 0  clingo -c n=10 p2.txt 0  clingo -c n=11 p2.txt 0  clingo -c n=12 p2.txt 0 |
| Output  of clingo | - |
| Answer  to Questions | Draw a table that lists the number of solutions and the times to compute all solutions. Use CPU time that clingo returns.   |  |  |  | | --- | --- | --- | | Value n | Number of solutions | time (in sec) | | 3 | 0 | 0.001 | | 4 | 2 | 0.001 | | 5 | 10 | 0.001 | | 6 | 4 | 0.001 | | 7 | 40 | 0.002 | | 8 | 92 | 0.004 | | 9 | 352 | 0.02 | | 10 | 724 | 0.1 | | 11 | 2680 | 0.79 | | 12 | 14200 | 9.25 | |

Problem 3

|  |  |
| --- | --- |
| Input  Program | % Define the Sudoku board.  1 {cell(R, C, V): V = 1..9} 1 :- R = 1..9, C = 1..9.  % Given initial values on the Sudoku board.  cell(1,1,8).  cell(2,3,7). cell(2,4,5). cell(2,9,9).  cell(3,2,3). cell(3,7,1). cell(3,8,8).  cell(4,2,6). cell(4,6,1). cell(4,8,5).  cell(5,3,9). cell(5,5,4).  cell(6,4,7). cell(6,5,5).  cell(7,3,2). cell(7,5,7). cell(7,9,4).  cell(8,6,3). cell(8,7,6). cell(8,8,1).  cell(9,7,8).  % No two cells on the same column can have the same value.  :- cell(R1, C, V), cell(R2, C, V), R1 != R2.  % No two cells on the same row can have the same value.  :- cell(R, C1, V), cell(R, C2 ,V), C1 != C2.  % No two cell in the same subgrid can have the same value.  :- cell(R1, C1, V), cell(R2, C2, V), (R1-1)/3 == (R2-1)/3, (C1-1)/3 == (C2-1)/3, R1 != R2, C1 != C2.  #show cell/3. |
| Command  Line | clingo ./program\_3.txt 0 |
| Output  of clingo | clingo version 5.4.0  Reading from ./problem\_3.txt  Solving...  Answer: 1  cell(1,2,9) cell(1,7,4) cell(2,3,8) cell(2,4,5) cell(2,8,1) cell(3,3,1) cell(3,8,6) cell(3,9,8) cell(4,4,1) cell(4,8,3) cell(5,5,4) cell(5,6,5) cell(5,7,7) cell(6,2,5) cell(6,6,7) cell(7,2,7) cell(7,5,9) cell(7,7,2) cell(8,3,3) cell(8,4,6) cell(9,1,8) cell(3,2,2) cell(2,2,3) cell(1,1,7) cell(2,1,4) cell(3,1,5) cell(1,3,6) cell(2,5,2) cell(1,4,3) cell(3,6,4) cell(2,6,6) cell(3,5,7) cell(1,5,1) cell(3,4,9) cell(1,6,8) cell(2,9,7) cell(3,7,3) cell(1,8,5) cell(2,7,9) cell(1,9,2) cell(4,1,2) cell(6,3,4) cell(5,2,6) cell(5,3,9) cell(6,1,1) cell(5,1,3) cell(4,2,8) cell(4,3,7) cell(6,5,3) cell(6,4,2) cell(4,5,6) cell(5,4,8) cell(4,6,9) cell(5,9,1) cell(5,8,2) cell(4,7,5) cell(6,9,6) cell(6,8,9) cell(6,7,8) cell(4,9,4) cell(9,2,1) cell(9,3,2) cell(8,2,4) cell(7,1,6) cell(8,1,9) cell(7,3,5) cell(8,6,2) cell(9,6,3) cell(7,4,4) cell(9,5,5) cell(8,5,8) cell(9,4,7) cell(7,6,1) cell(9,8,4) cell(8,9,5) cell(8,8,7) cell(9,9,9) cell(8,7,1) cell(9,7,6) cell(7,8,8) cell(7,9,3)  SATISFIABLE  Models : 1  Calls : 1  Time : 0.017s (Solving: 0.00s 1st Model: 0.00s Unsat: 0.00s)  CPU Time : 0.017s |

Problem 4

|  |  |
| --- | --- |
| Input  Program | {cell(X,Y,N): X=1..16, Y=1..16, X1<=X, X<=X1+3, Y1<=Y, Y<=Y1+3} = 1 :- N=1..16, X1 = 4\*(0..3)+1, Y1 = 4\*(0..3)+1.  :- cell(X,Y,N), cell(X,Y,N1), N1!=N.  :- cell(X,Y,N), cell(X,Y1,N), Y1!=Y.  :- cell(X,Y,N), cell(X1,Y,N), X1!=X.  %Instance  cell(1,1,9).  cell(1,2,14).  cell(1,6,3).  cell(1,8,5).  cell(1,9,15).  cell(1,11,2).  cell(1,15,7).  cell(1,16,1).  cell(2,1,6).  cell(2,2,12).  cell(2,6,14).  cell(2,11,10).  cell(2,15,5).  cell(2,16,11).  cell(3,1,4).  cell(3,4,7).  cell(3,5,6).  cell(3,8,13).  cell(3,9,16).  cell(3,12,1).  cell(3,13,2).  cell(3,16,9).  cell(4,2,15).  cell(4,3,16).  cell(4,5,9).  cell(4,6,7).  cell(4,11,11).  cell(4,12,6).  cell(4,14,3).  cell(4,15,14).  cell(5,2,7).  cell(5,3,15).  cell(5,14,2).  cell(5,15,16).  cell(6,1,5).  cell(6,3,13).  cell(6,5,14).  cell(6,7,15).  cell(6,10,10).  cell(6,12,3).  cell(6,14,1).  cell(6,16,8).  cell(7,2,8).  cell(7,4,10).  cell(7,6,9).  cell(7,7,4).  cell(7,8,11).  cell(7,9,13).  cell(7,10,6).  cell(7,11,15).  cell(7,13,14).  cell(7,15,3).  cell(8,1,16).  cell(8,5,5).  cell(8,7,3).  cell(8,10,14).  cell(8,12,9).  cell(8,16,6).  cell(9,1,15).  cell(9,5,16).  cell(9,7,10).  cell(9,10,9).  cell(9,12,13).  cell(9,16,14).  cell(10,2,9).  cell(10,4,6).  cell(10,6,5).  cell(10,7,13).  cell(10,8,3).  cell(10,9,1).  cell(10,10,15).  cell(10,11,4).  cell(10,13,7).  cell(10,15,12).  cell(11,1,2).  cell(11,3,8).  cell(11,5,15).  cell(11,7,14).  cell(11,10,16).  cell(11,12,12).  cell(11,14,5).  cell(11,16,13).  cell(12,2,13).  cell(12,3,12).  cell(12,14,9).  cell(12,15,11).  cell(13,2,5).  cell(13,3,3).  cell(13,5,2).  cell(13,6,16).  cell(13,11,13).  cell(13,12,10).  cell(13,14,12).  cell(13,15,9).  cell(14,1,8).  cell(14,4,4).  cell(14,5,12).  cell(14,8,1).  cell(14,9,6).  cell(14,12,7).  cell(14,13,15).  cell(14,16,3).  cell(15,1,10).  cell(15,2,1).  cell(15,6,15).  cell(15,11,16).  cell(15,15,6).  cell(15,16,2).  cell(16,1,11).  cell(16,2,2).  cell(16,6,8).  cell(16,8,14).  cell(16,9,3).  cell(16,11,1).  cell(16,15,10).  cell(16,16,7). |
| Command  Line | clingo ./program\_4.txt 0 |
| Output  of clingo | clingo version 5.4.0  Reading from ./problem\_4.txt  Solving...  Answer: 1  cell(1,1,9) cell(1,2,14) cell(1,6,3) cell(1,8,5) cell(1,9,15) cell(1,11,2) cell(1,15,7) cell(1,16,1) cell(2,1,6) cell(2,2,12) cell(2,6,14) cell(2,11,10) cell(2,15,5) cell(2,16,11) cell(3,1,4) cell(3,4,7) cell(3,5,6) cell(3,8,13) cell(3,9,16) cell(3,12,1) cell(3,13,2) cell(3,16,9) cell(4,2,15) cell(4,3,16) cell(4,5,9) cell(4,6,7) cell(4,11,11) cell(4,12,6) cell(4,14,3) cell(4,15,14) cell(5,2,7) cell(5,3,15) cell(5,14,2) cell(5,15,16) cell(6,1,5) cell(6,3,13) cell(6,5,14) cell(6,7,15) cell(6,10,10) cell(6,12,3) cell(6,14,1) cell(6,16,8) cell(7,2,8) cell(7,4,10) cell(7,6,9) cell(7,7,4) cell(7,8,11) cell(7,9,13) cell(7,10,6) cell(7,11,15) cell(7,13,14) cell(7,15,3) cell(8,1,16) cell(8,5,5) cell(8,7,3) cell(8,10,14) cell(8,12,9) cell(8,16,6) cell(9,1,15) cell(9,5,16) cell(9,7,10) cell(9,10,9) cell(9,12,13) cell(9,16,14) cell(10,2,9) cell(10,4,6) cell(10,6,5) cell(10,7,13) cell(10,8,3) cell(10,9,1) cell(10,10,15) cell(10,11,4) cell(10,13,7) cell(10,15,12) cell(11,1,2) cell(11,3,8) cell(11,5,15) cell(11,7,14) cell(11,10,16) cell(11,12,12) cell(11,14,5) cell(11,16,13) cell(12,2,13) cell(12,3,12) cell(12,14,9) cell(12,15,11) cell(13,2,5) cell(13,3,3) cell(13,5,2) cell(13,6,16) cell(13,11,13) cell(13,12,10) cell(13,14,12) cell(13,15,9) cell(14,1,8) cell(14,4,4) cell(14,5,12) cell(14,8,1) cell(14,9,6) cell(14,12,7) cell(14,13,15) cell(14,16,3) cell(15,1,10) cell(15,2,1) cell(15,6,15) cell(15,11,16) cell(15,15,6) cell(15,16,2) cell(16,1,11) cell(16,2,2) cell(16,6,8) cell(16,8,14) cell(16,9,3) cell(16,11,1) cell(16,15,10) cell(16,16,7) cell(12,1,1) cell(2,3,1) cell(8,4,1) cell(7,3,2) cell(4,4,2) cell(5,1,3) cell(9,2,3) cell(2,4,3) cell(11,2,4) cell(8,3,4) cell(3,3,5) cell(9,4,5) cell(6,2,6) cell(16,3,6) cell(13,1,7) cell(9,3,7) cell(1,4,8) cell(15,3,9) cell(6,4,9) cell(3,2,10) cell(10,3,10) cell(8,2,11) cell(1,3,11) cell(11,4,11) cell(7,1,12) cell(16,4,12) cell(4,1,13) cell(15,4,13) cell(10,1,14) cell(14,3,14) cell(5,4,14) cell(13,4,15) cell(14,2,16) cell(12,4,16) cell(7,5,1) cell(9,6,1) cell(4,7,1) cell(8,6,2) cell(2,7,2) cell(12,8,2) cell(15,5,3) cell(2,5,4) cell(12,6,4) cell(15,8,4) cell(14,7,5) cell(11,6,6) cell(5,7,6) cell(13,8,6) cell(12,5,7) cell(15,7,7) cell(8,8,7) cell(5,5,8) cell(12,7,8) cell(4,8,8) cell(16,7,9) cell(11,8,9) cell(1,5,10) cell(14,6,10) cell(5,8,10) cell(10,5,11) cell(3,6,11) cell(13,7,11) cell(6,6,12) cell(3,7,12) cell(9,8,12) cell(16,5,13) cell(5,6,13) cell(2,8,15) cell(1,7,16) cell(6,8,16) cell(5,10,1) cell(6,9,2) cell(14,10,2) cell(10,12,2) cell(3,10,3) cell(11,11,3) cell(5,9,4) cell(16,10,4) cell(1,12,4) cell(4,9,5) cell(12,10,5) cell(5,11,5) cell(15,12,5) cell(12,11,6) cell(11,9,7) cell(2,10,7) cell(6,11,7) cell(8,9,8) cell(13,10,8) cell(9,11,8) cell(2,12,8) cell(2,9,9) cell(14,11,9) cell(12,9,10) cell(9,9,11) cell(15,10,11) cell(5,12,11) cell(15,9,12) cell(4,10,12) cell(8,11,12) cell(1,10,13) cell(13,9,14) cell(3,11,14) cell(12,12,14) cell(16,12,15) cell(7,12,16) cell(13,13,1) cell(11,15,1) cell(9,15,2) cell(12,13,3) cell(4,13,4) cell(9,14,4) cell(6,15,4) cell(13,16,4) cell(16,13,5) cell(7,16,5) cell(9,13,6) cell(1,14,6) cell(7,14,7) cell(15,13,8) cell(10,14,8) cell(3,15,8) cell(5,13,9) cell(11,13,10) cell(8,14,10) cell(4,16,10) cell(6,13,11) cell(14,14,11) cell(1,13,12) cell(5,16,12) cell(8,13,13) cell(2,14,13) cell(14,15,13) cell(15,14,14) cell(3,14,15) cell(8,15,15) cell(12,16,15) cell(2,13,16) cell(16,14,16) cell(10,16,16)  SATISFIABLE  Models : 1  Calls : 1  Time : 0.103s (Solving: 0.00s 1st Model: 0.00s Unsat: 0.00s)  CPU Time : 0.103s |

Problem 5

|  |  |
| --- | --- |
| Input  Program |  |
| Command  Line | clingo ./program\_5.txt 0 |
| Output  of clingo |  |

Problem 6

|  |  |
| --- | --- |
| Input  Program |  |
| Command  Line |  |
| Output  of clingo |  |

Problem 7

|  |  |
| --- | --- |
| Input  Program |  |
| Command  Line |  |
| Output  of clingo |  |

Problem 8

|  |  |
| --- | --- |
| Input  Program |  |
| Command  Line | You should write multiple command lines below. |
| Output  of clingo |  |
| Answer  to Questions | Draw a table that lists the maximum value of bishops when the chessboard is n by n, where n is 3, 4, 5, 6, 7, 8. Infer the general function f(n) that returns the maximum value of bishops.   |  |  | | --- | --- | | Value n | f(n) | | 3 |  | | 4 |  | | 5 |  | | 6 |  | | 7 |  | | 8 |  |   f(n) = |

Problem 9

|  |  |
| --- | --- |
| Input  Program |  |
| Command  Line | You should write multiple command lines below. |
| Output  of clingo |  |
| Answer  to Questions | Fill in the values accordingly.   |  |  | | --- | --- | | Exact value of A(1) |  | | Exact value of A(2) |  | | Exact value of A(3) |  | | Largest lower bound for A(4)  Note: it would take longer time when you increase the value of n. Thus, you may stop increasing the value of n when your program does not terminate within 10 minutes and submit the last trial of n. |  | |