

Department of Computer Science
CS 2209A — Applied Logic for Computer Science

Assignment 4

Assigned: 19 November 2018

Due: 3 December 2018 (on OWL before midnight)

1 Part I

1. Write a Prolog function *last(List, LastItem)* that gives the last item in a list. Use your function on the following lists: [], [a], [a,b,c]. Using comments in your program, provide a short explanation of what the program is doing to do this computation.
2. The lecture notes gave a representation for the 8-Queens problem. SWI-Prolog has a permutation function. Given the following Prolog function for the 8-queens program:

`eightQueens(Board) :- permutation([1,2,3,4,5,6,7,8], Board), checkDiagonals(Board).`

- (a) Complete the program with a function to perform the `checkDiagonals` function. Run your program.
- (b) Using comments in your program, provide a short explanation of your method to check the diagonals on the board and how your function implements this method.
- (c) Add another parameter, *N*, to your function and generalize your previous answer to make an *N*-Queens function. Note: SWI-Prolog has a function *numlist* that generates a list `[1, ..., N]`. Run your new function with *N* having the values 1 to 8.

2 Part II

1. A basic modelling technique in ASP is the generate-and-test methodology. Following on the *N*-Queens problem in the previous question, the following “generate” portion of an ASP program represents and generates board positions for the *N*-Queens problem. Here exactly *n* queens are placed in board positions (*i*, *j*) using the statement given. Gringo grounds this statement in all possible ways without creating duplicates. This “generate” portion obviously over-generates possible solutions. The `#show` command simply controls the output.

```
n {queen(1..n, 1..n)} n
#show queen/2.
```

少了一个点!!!

Write the constraints that perform the “test” part of the ASP program in order to eliminate the incorrectly generated board positions so that the ASP program produces only the correct solutions for the *N*-Queens problem. Using comments in your program, provide a short explanation of what the constraints are doing to eliminate the incorrect board positions.

Call your program “queens.lp”. To run the program use:

```
gringo queens.lp --const n=N | clasp 0
```

where *N* is replaced with an integer. Run your program with the integers from 1 to 8.