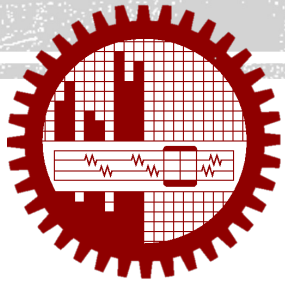
CSE-318
OFFLINE-2 ON CSP

Dr. M A Nayeem

Assistant Professor

CSE, BUET



OUTLINE

- Tasks
- Recap
- Tips

tips optional

10 by 10
soduku er reduced version
ekta row te same digit 2 bar na
ekta col te same digit 2 bar na

LATIN SQUARE

0	4	8	2	3	9	6	7	1	5
3	6	2	8	7	1	9	5	0	4
8	9	3	1	0	6	4	2	5	7
1	7	6	5	4	8	0	3	2	9
2	1	9	0	6	7	5	8	4	3
5	2	7	4	9	3	1	0	8	6
4	3	0	6	1	5	2	9	7	8
9	8	5	7	2	0	3	4	6	1
7	0	1	9	5	4	8	6	3	2
6	5	4	3	8	2	7	1	9	0

LATIN SQUARE COMPLETION PROBLEM

```
N=10;
start=
[|
0, 0, 6, 0, 0, 3, 4, 0, 10, 0 |
2, 6, 4, 0, 0, 0, 0, 0, 9, 0 |
0, 2, 10, 0, 0, 0, 0, 0, 5, 9 |
10, 1, 5, 4, 2, 0, 0, 0, 0, 0 |
0, 0, 0, 0, 1, 9, 8, 4, 0, 0 |
0, 0, 3, 2, 9, 0, 0, 1, 0, 0 |
6, 0, 0, 0, 0, 7, 0, 10, 0, 5 |
0, 0, 0, 0, 0, 8, 6, 5, 0, 7 |
1, 3, 0, 6, 0, 0, 5, 0, 0, 2 |
0, 5, 0, 9, 6, 2, 0, 0, 8, 0 |];
```

0 mane empty
amader fillup korte hobe

1 theke 10 porjonto valid value

TASK: IMPLEMENT TWO SOLVERS

- Simple Backtracking (BT)
- Forward Checking (FC)

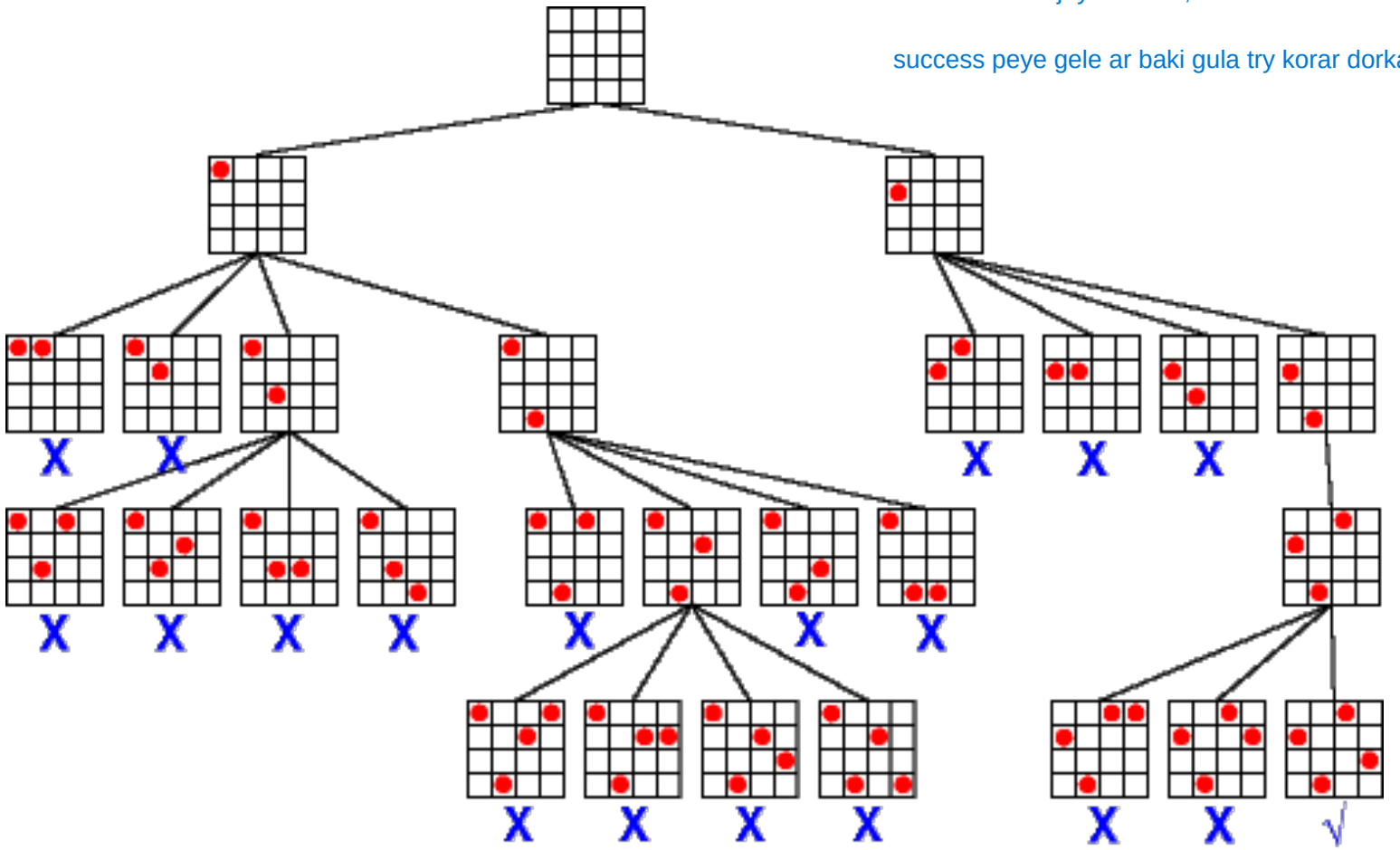
RECAP: 4-QUEENS AND BT

BT mane back tracking

1st var holo 1st col e 1st queen ta koi boshbe
2nd var holo 2nd col e 2nd queen ta koi boshbe
.
.

X mane ar jeye luv nai, so backtrack korbo

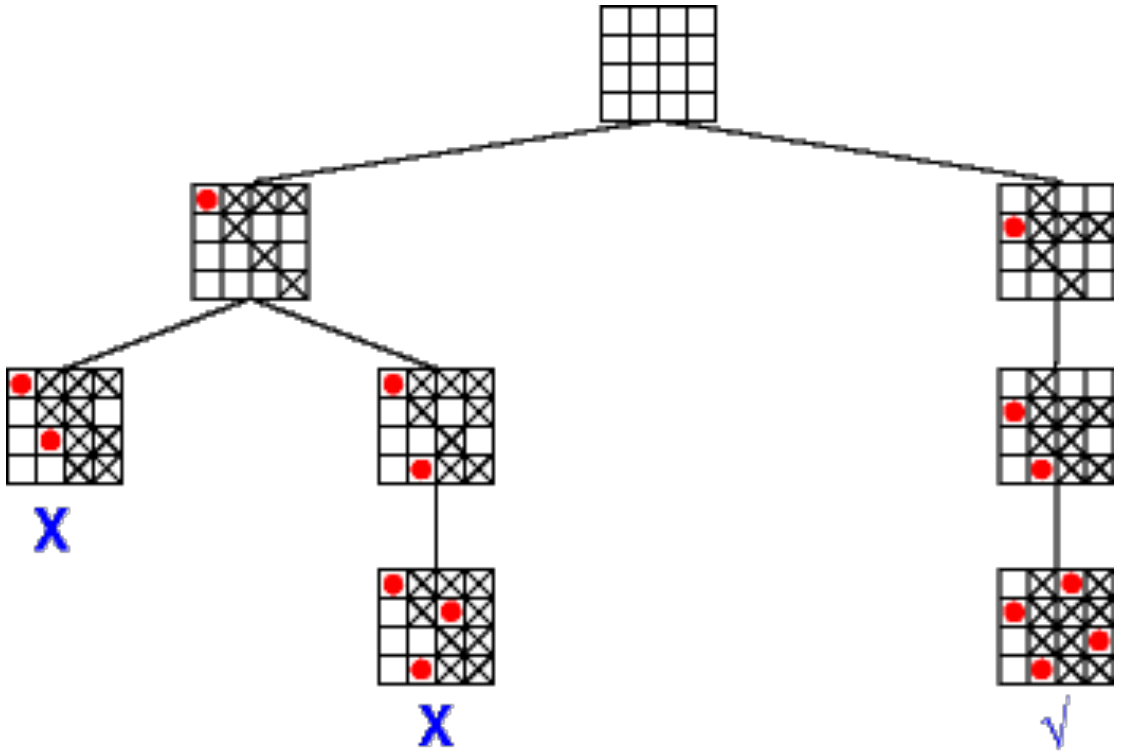
success peye gele ar baki gula try korar dorkar nai



forward checking holo ager tar e ektu advanced version

1st col er 4tai hocche 1st variable er domain, mane 1st variable ta kon kon jaygay boshte pare ta
ekhon 1st ghor fill korle seoncd er jonno domain ashole 4 ta na hoye 2 ta hoye jay

RECAP: 4-QUEENS AND FC



4st var er domain
empty hoye gelo bole
ar agabo na

Eita russel er boi theke tule diche

RECAP: SOLVER

emon variable age choose korbo je amake failure er dike niye jabe

```
function BACKTRACKING-SEARCH(csp) returns a solution or failure
    return BACKTRACK(csp, { })

function BACKTRACK(csp, assignment) returns a solution or failure
    if assignment is complete then return assignment
    var ← SELECT-UNASSIGNED-VARIABLE(csp, assignment)
    for each value in ORDER-DOMAIN-VALUES(csp, var, assignment) do
        if value is consistent with assignment then
            add {var = value} to assignment
            inferences ← INFERENCE(csp, var, assignment)
            if inferences ≠ failure then
                add inferences to csp
                result ← BACKTRACK(csp, assignment)
                if result ≠ failure then return result
                remove inferences from csp
            remove {var = value} from assignment
    return failure
```

surute kichu kori nai bole empty thakbe, left er ta hocche shei recursive function

var ordering heuristic

value ordering heuristic

prothom ta valo vabe korte hobe

second ta amader iccha moto

TASK: 5 VARIABLE ORDERING HEURISTICS

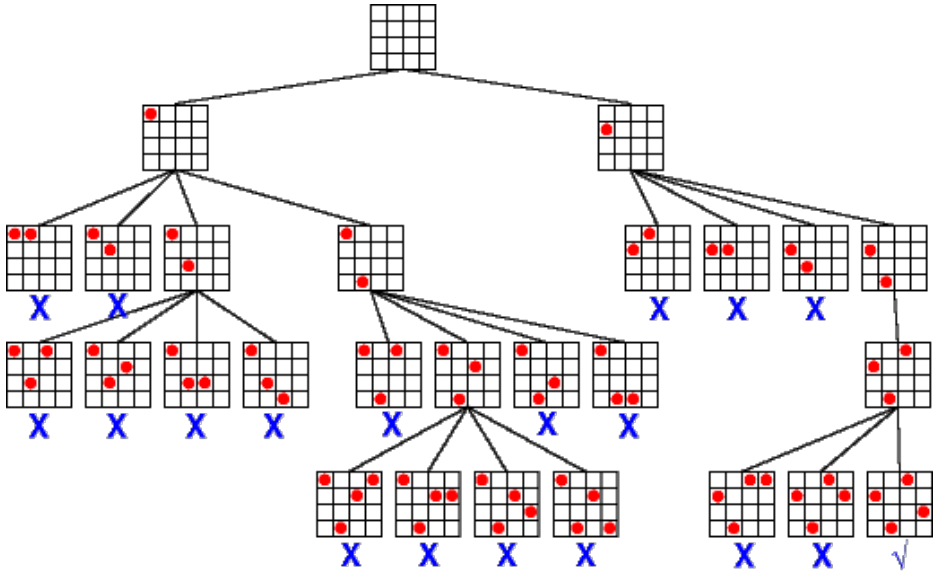
- VAH1
  - The variable chosen is the one with the smallest domain
- VAH2
  - The variable chosen is the one with the maximum degree to unassigned variables. Also, called max-forward-degree
- VAH3
  - The variable chosen by VAH1, Ties are broken by VAH2 second tateo tie hole random
- VAH4
  - The variable chosen is the one that minimizes the VAH1 / VAH2 ratio
- VAH5
  - A random unassigned variable is chosen

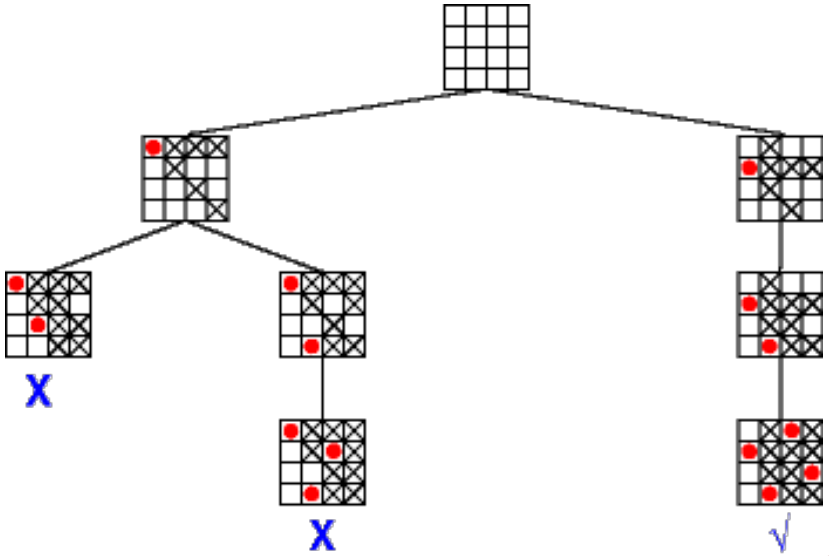
TASK: 1 VALUE ORDERING HEURISTICS

- Your choice
  - Justification -> Offline-2 Report

TASK: PERFORMANCE MEASURE

- How to compare among solutions schemes?
  - Number of total node
  - Number of backtracks
  - Runtime





TASK: REPORT (SUBMIT WITH CODE)

- Value Order Heuristic
  - Justify your choice
- Table: Summarizes all results
  - 5 problems, 2 solvers, 5 VAH
  - Mark the best (optionally 2nd best) scheme for each solver
- Conclusion
  - Which scheme seems the best according to your opinion?
  - Provide justification as much as possible

Problem	Solver	VAH	#Node	#BT	Runtime
d-10-01	BT	VAH1			
	BT	VAH2			
	BT	VAH3			
	BT	VAH4			
	BT	VAH5			
	FC	VAH1			
	FC	VAH2			
	FC	VAH3			
	FC	VAH4			
	FC	VAH5			
…	…	…	…	…	

TIPS: AN OOP DESIGN [OPTIONAL]

- *Class* Variable
  - Domain: list of values
- Assignment var class ei, setvalue nam dilam
  - *Hashmap*: variable -> value
- *Class* Constraint
  - Scope: a tuple of variables
  - Condition: Boolean function to be applied to scope
  - *method* holds(assignment) -> bool
- *Class* CSP
  - Variables
  - Constraints

- *Class* Variable_Order_Heuristic
  - *method* get_next_variable(csp, assignment) -> variable
- *Class* CSP_Solver
  - Variable_Order_Heuristic
  - CSP = null, assignment = null
  - *method* solve(csp, assignment) -> solution
  - *method* value_order_heuristic(variable) -> value

TIPS: AUTOMATE VIA SCRIPT [OPTIONAL]

- Command line arguments

- Your script:
  - ./yourProgram data=# solver=# vah=# …
  - …
  - …
  - …

ACKNOWLEDGEMENT

- Special thanks to
  - Abu Wasif Sir
  - Hasibul Hisham
  - Md. Ashraful Islam
  - Tanjeem Azwad Zaman

THANKS
KEEP SMILING :)