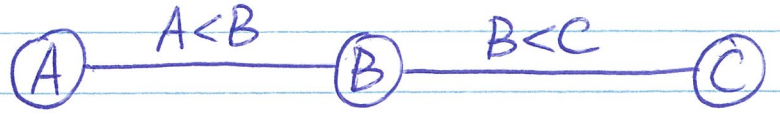
Arc Consistency Example Lecture 6 — Alice Gao.



dom(A) = {1, 2, ~~3~~ ④, ~~4~~ ①}

dom(B) = {~~1~~ ③, 2, 3, ~~4~~ ②}

dom(C) = {~~1~~, ~~2~~ ⑤, 3, 4}

queue: ~~(A, A<B)~~ ①, ~~(B, A<B)~~ ③, ~~(B, B<C)~~ ②, ~~(C, B<C)~~ ⑤, ~~(A, A<B)~~ ④

① remove (A, A<B).
remove 4 from dom(A).
add nothing to queue. (why not add back (B, A<B)?)

② remove (B, B<C)
remove 4 from dom(B)
add (A, A<B) back into queue.

③ remove (B, A<B).
remove 1 from dom(B)
add (C, B<C) into queue (already there).

④ remove (A, A<B).
remove 3 from dom(A).
add nothing to queue.

⑤ remove (C, B<C).
remove 1 and 2 from dom(C).
add nothing to queue.

Arc Consistency Example Lecture 6.

Alice Gao

When we remove (A, A<B) and reduce A's domain, should we add (B, A<B) back into the queue?

We don't add back arcs involving the same constraint and a different variable.

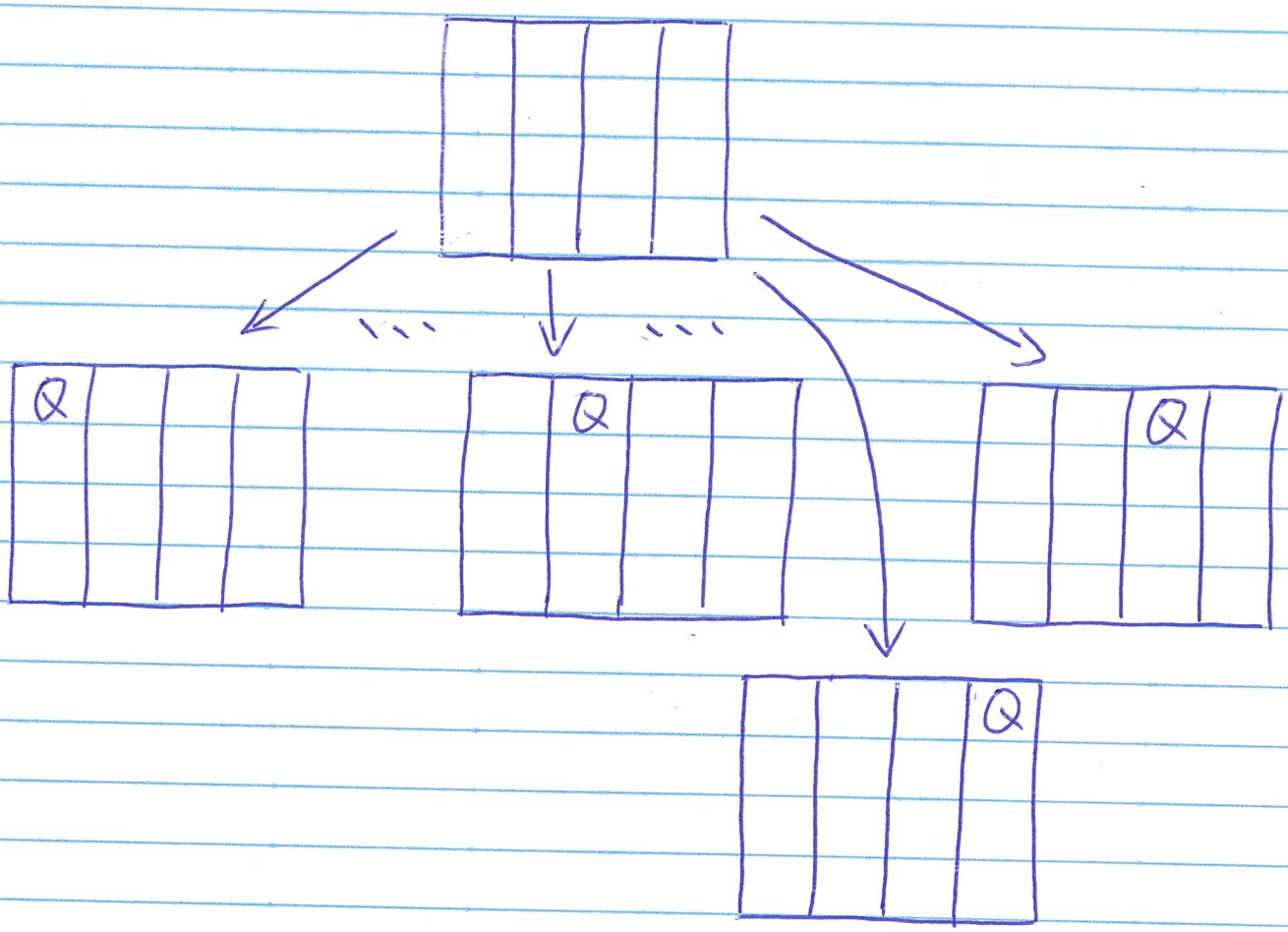
- If (B, A<B) was not consistent before, it was in the queue. No need to add it again.

- If (B, A<B) was consistent before, We will show that (B, A<B) is still consistent after.

Proof by Contradiction:

- Suppose (B, A<B) is not consistent after.
- A value of B could only satisfy the constraint if A takes the dropped value.
- But we dropped the value of A because there were no value of B that ~~allow~~ satisfy the constraint when A takes the dropped value.
  — contradiction.

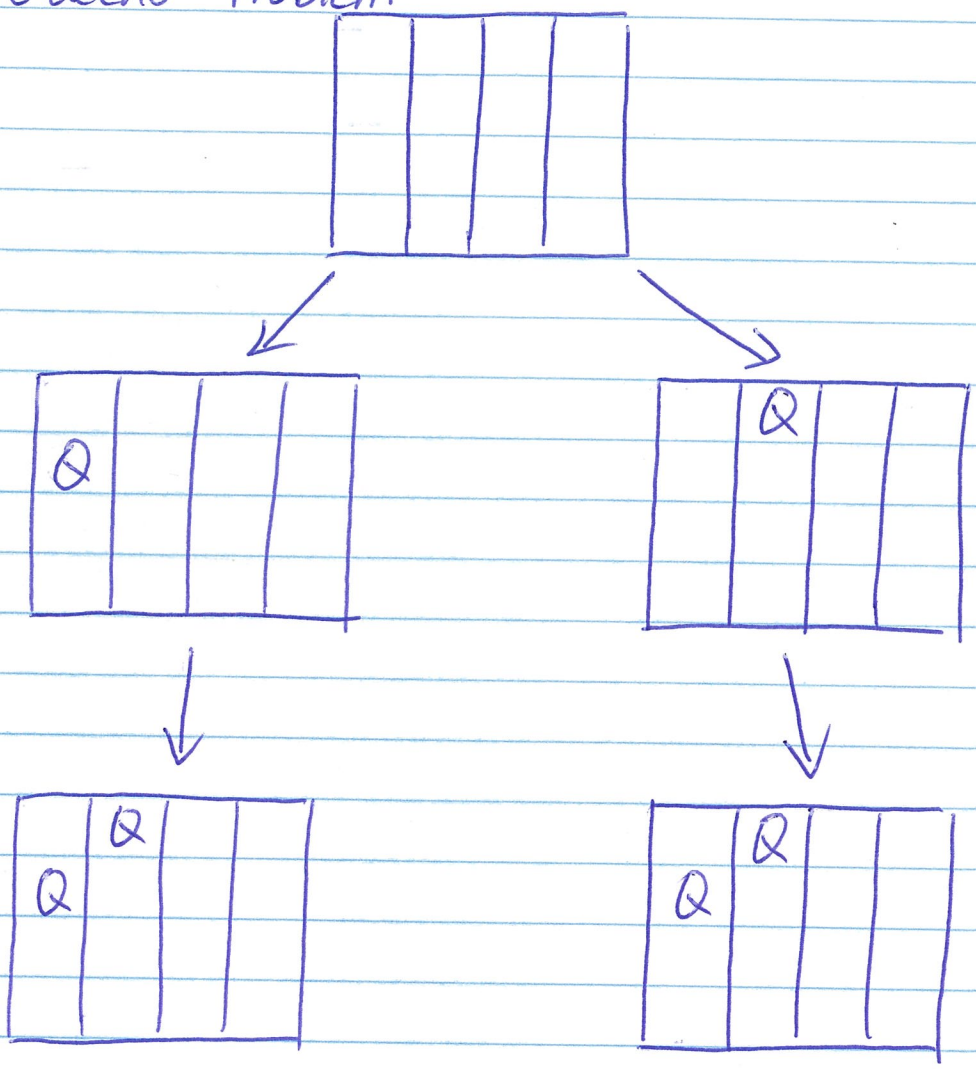
Four-Queens Problem.



- Count the # of successors.
- Count the # of leaf nodes in the search tree. A
  4-queen board.
- Count the # of unique four-queen boards. B

Why is A much larger than B?

Four-Queens Problem



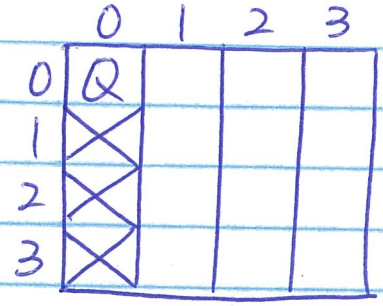
- CSP is commutative.
  Assigning values to variables in different orders will arrive at the same state.

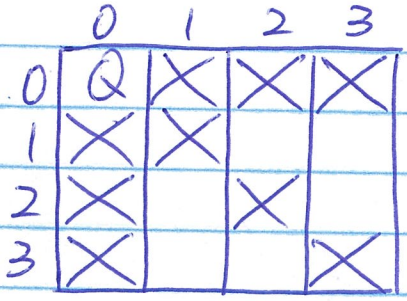
- Consider one variable when generating successors

Backtracking Search Lecture 6.

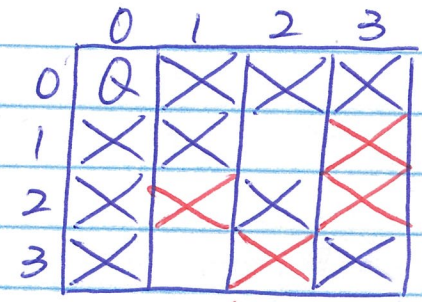
Alice Gao

Example 1:



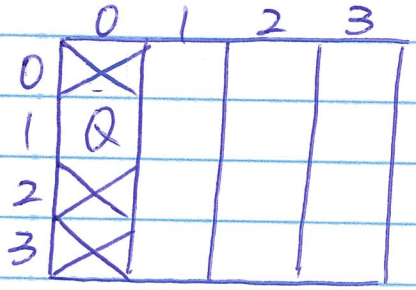


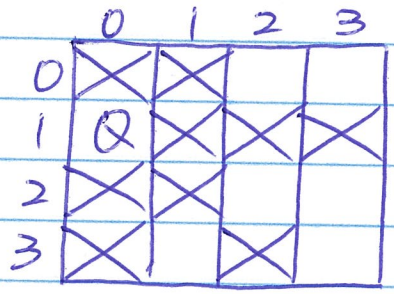
forward checking



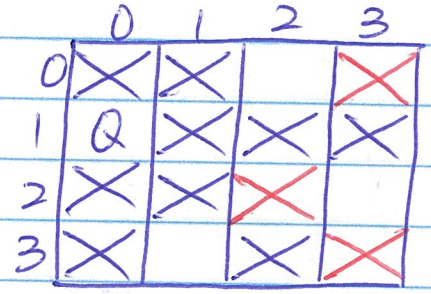
no solution.
maintaining arc consistency.

Example 2:





forward checking.



maintaining arc consistency.