Algoritma Backtracking:

1. Inisialisasi nilai awal untuk kolom nol dan baris

pertama.

2. Input bilangan integer untuk menentukan ukuran

papan catur.

3. Periksa apakah posisi baris lebih besar dari nol:

a. Jika posisi baris > 0, maka pindah ke kolom berikutnya dan lanjutkan

langkah 4.

b. Jika posisi baris < 0, maka kondisi tidak terpenuhi dan akan keluar dari

perulangan sehingga algoritma selesai.

4. Periksa apakah posisi kolom lebih kecil atau sama dengan ukuran papan dan periksa juga apakah posisi kolom dan diagonal yang sama dari queen sudah terisi oleh queen yang lain:

a. Jika salah satu kondisi tidak terpenuhi maka

kembali ke langkah 5.

b. Jika kedua kondisi terpenuhi, maka pindah ke

kolom selanjutnya dan lakukan kembali ke

langkah 4.

5. Periksa apakah posisi kolom kurang dari atau sama

dengan ukuran papan:

a. Jika memenuhi, cek apakah queen telah

ditempatkan pada seluruh baris. Jika ya, maka

cetak queen dan algoritma selesai. Jika tidak,

maka pindah ke baris selanjutnya pada kolom

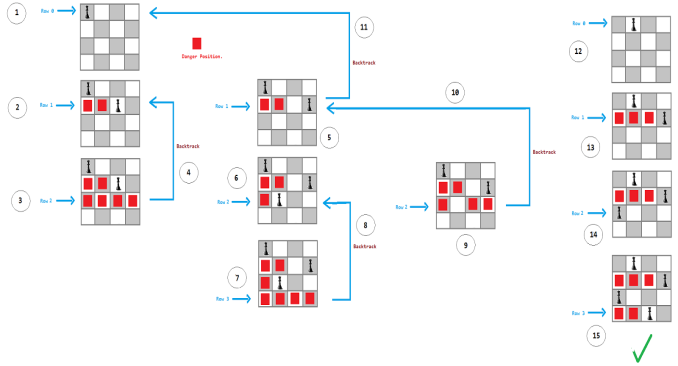
awal.

b. Jika tidak memenuhi kondisi, maka kembali ke

baris sebelumnya.

6. Ulangi langkah 3.

Contoh Proses Backtracking:



Untuk penyimpanannya tidak di perkecil karena untuk pencarian 8 queen problem, diperlukan array 2D untuk menentukan lokasi queen dengan data bolean.

Logic :

1) mulai dari colom paling kiri

2) If (semua queen sudah ada di ataas papan )

    return true

3)While (mencoba coba semua row di kolom tersebut)

    a) jika queen berada dalam lokasi yang aman, tandai lokasi sekarang [row,colom] sebagai bagian dari solution

    b) jika menembatkan queen di [row,column] tidak bertabrakan dengan queen lain, maka return true.

    c) jika lokasi penempatan queen bertabrakan, maka hapus tanda lokasi [row,column] (Backtrack) dan kembali ke step (a) untuk mencari jalan lain.

4) jika semua row sudah di coba dan tidak ada yang berhasil, return false dan kembali ke langkah sebelumnya