

Knight Tour

05 February 2022
23:07

Why we are marking one move inside base case

The image shows a handwritten diagram on the left and a code snippet on the right. The diagram illustrates the recursive calls for the Knight Tour problem, showing the sequence of moves and the state of the chessboard. The code snippet is a Java method `printKnightsTour` that takes a chessboard, row, column, and upcoming move as input. It checks for base cases (row out of bounds, column out of bounds, or cell already occupied) and marks the current cell with the upcoming move value. The code also shows recursive calls for the next moves.

```
public static void printKnightsTour(int[][] chess, int r, int c, int upcomingMove) {
    if (r < 0 || c < 0 || r >= chess.length || c >= chess[0].length)
        return;
    chess[r][c] = upcomingMove;

    printKnightsTour(chess, r-2, c+1, upcomingMove+1);
    printKnightsTour(chess, r-1, c+2, upcomingMove+1);
    printKnightsTour(chess, r+1, c+2, upcomingMove+1);
    printKnightsTour(chess, r+2, c+1, upcomingMove+1);
    printKnightsTour(chess, r+2, c-1, upcomingMove+1);
    printKnightsTour(chess, r+1, c-2, upcomingMove+1);
    printKnightsTour(chess, r-1, c-2, upcomingMove+1);
    printKnightsTour(chess, r-2, c-1, upcomingMove+1);

    chess[r][c] = 0;
}
```

Dekho jab hum 1st call lgane hai tb tak chess pe 0 block mark hote hai

Jab hum 2nd call lgane jaate haitab tak 1 element mark hota hai

To isi trh jab hum 25th call lgane jaayenge us time pe total 24 element marke honge aur 25th hume khud mark krna pdega

Ya fir aise smjh lo ki pahle base case check hota hai uske baad marking hoti hai

To jb upcomingMove ki value 1 thi base case check hua tab tak 0 block hi mark tha

Fir hum mark krte hai

Aur next call lgti hai upcomingMove ki value 2 ho hai base case check hota hai tab tak 1 block hi mark tha

To isi trh jab upcomingMove ki value 25 hogi tab tak sirf 24 blocks hi mark honge to bcha hua 1 hume khud mark krna pdega

Hum base case me hi +1 kyu nhi kr dete ?

```
if(r<0 || c<0 || r>=chess.length || c>=chess[0].length || chess[r][c]!=0){
    return;
}

if(upcomingMove==chess.length*chess[0].length+1){
    // chess[r][c]=upcomingMove;
    displayBoard(chess);
    // chess[r][c]=0;
    return;
}

chess[r][c] = upcomingMove;

printKnightsTour(chess, r-2, c+1, upcomingMove+1);
printKnightsTour(chess, r-1, c+2, upcomingMove+1);
printKnightsTour(chess, r+1, c+2, upcomingMove+1);
printKnightsTour(chess, r+2, c+1, upcomingMove+1);
printKnightsTour(chess, r+2, c-1, upcomingMove+1);
printKnightsTour(chess, r+1, c-2, upcomingMove+1);
printKnightsTour(chess, r-1, c-2, upcomingMove+1);
printKnightsTour(chess, r-2, c-1, upcomingMove+1);

chess[r][c]=0;
```

Dekho agr hum base condition me hi +1 add kr dete hai to isse ye problem hogi

Jab upcomingMove ki value 25 hogi tab base case hit nhi hoga aur niche jaakr mark ho jaayega lekin problem suru hoti hai ab

Ab hoga ye ki upcomingMove bdakar hum call lgane ki koshish krnege lekin hmesha return hi hoga kyunki chess[r][c] !=0 kisi bhi halat me false nhi hogi kyunki saare blocks marked hai aur kisi pe bhi 0 nhi hai

To ab saari 8 calls chalne ke baad jo 25th wala block mark hua tha wo firse unmark ho jaayega kyunki code ab backtrack krne lg jaayega that means ki hmara base case kabhi hit nhi hoga aur koi ans print nhi hoga