Pseudo Code of n\_KNIGHT\_problem using Genetic Algorithm

Declaration of arrays like boards(3D array), chromosome(2D array),fitness(1D array) and sol

Initialization of some macros \_\_BOARD\_SIZE\_\_ to 8 (or any number whose answer is required), \_\_EMPTY\_\_ to ‘ ‘ , \_\_KNIGHT\_\_ to ‘O’, \_\_ATTACK\_\_ to ‘X’, \_\_POP\_\_ to 200, \_\_MAX\_KNIGHT\_\_ to a number which would calculate from the formula 2\*(((\_\_BOARD\_SIZE\_\_ \* \_\_BOARD\_SIZE\_\_)/9)+4)

Declaration and Initialization of variable like nParents to \_\_POP\_\_/2 and trying knights to ((\_\_BOARD\_SIZE\_\_ \* \_\_BOARD\_SIZE\_\_)/9)+4,isOver to 0, iter to 0 and maxIter to 5000

Initialize all the elements of chromosome(2D array) to a random number excluding the number which will result in placing the knight in first or last column or first row or last row

And make sure no 2 elements in 1D array are same

Reset all the elements of boards(3D array) to \_\_EMPTY\_\_

Convert random numbers in chromosome array to row and column by dividing and taking reminder operator on that number and place there knight

Place attacks of each knight by giving relative coordinates of knights (2,1),(2,-1), (-2,1),(-2,-1), (1,2),(1,-2), (-1,2)and(-1,-2)

Calculate No. of spaces in each board and put that in fitness array

Sort in ascending order the fitness and also sort the chromosome array on the basis of fitness

Check if first element of fitness is 0

If Yes

Print Board

Else

Select the first half of the population as parents

Give even childrens the even values of even parents

Give even childrens the odd values of odd parents

Give odd childrens the odds values of even parents

Give odd childrens the even values of odd parents

Give random value at ran(1,2),(1,-2)dom position to every chromosome element by making sure that the random may not result in placing knight first row or last row or first column or last columns and no two values in that array are the same.

Add one plus value to iter variable

Repeat the process just after randome initialization of chromosome array until iter become equal to maxIter

Print “Solution not found”