REGNO: 2021506076

SUDOKU SOLVER

NAME :Y RUDHRA

TOPIC : SUDOKU SOLVER

PROBLEM STATEMENT:

Sudoku is a popular game originated in Japan and has become a craze in India of late. Sudoku is a logic-based, number-placement puzzle. The objective is to fill a 9 × 9 grid with digits so that each column, each row, and each of the nine 3 × 3 sub-grids contain all ofthe digits from 1 to 9.

The purpose of the project is to develop more effective algorithm in order to reduce the computing time and human efforts. This algorithm is similar to human methods. This project implements the pencil-and-paper algorithm in C programming language using data structures.

DATA STRUCTURE TO BE USED:

1. **Linked List** : Having given a set of input values and empty squares of no of blocks of different layers, a no between 1to 9 is to be dropped in the empty squares to check whether it is valid. So each possible no will be traversed through the available empty squares.
2. **Queue**: We use undo to roll back to the previous state.

ARCHITECTURE DIAGRAM:

We follow 3 basic steps to arrive at the solution for the Sudoku puzzle in Pencil and paper algorithm.

1) Naked singles method
2) Hidden singles method
3) Brute force method

NAKED SINGLES METHOD:

It is used when the row, column and box hold 8 different numbers and one single number is left for that square.

HIDDEN SINGLES METHOD:

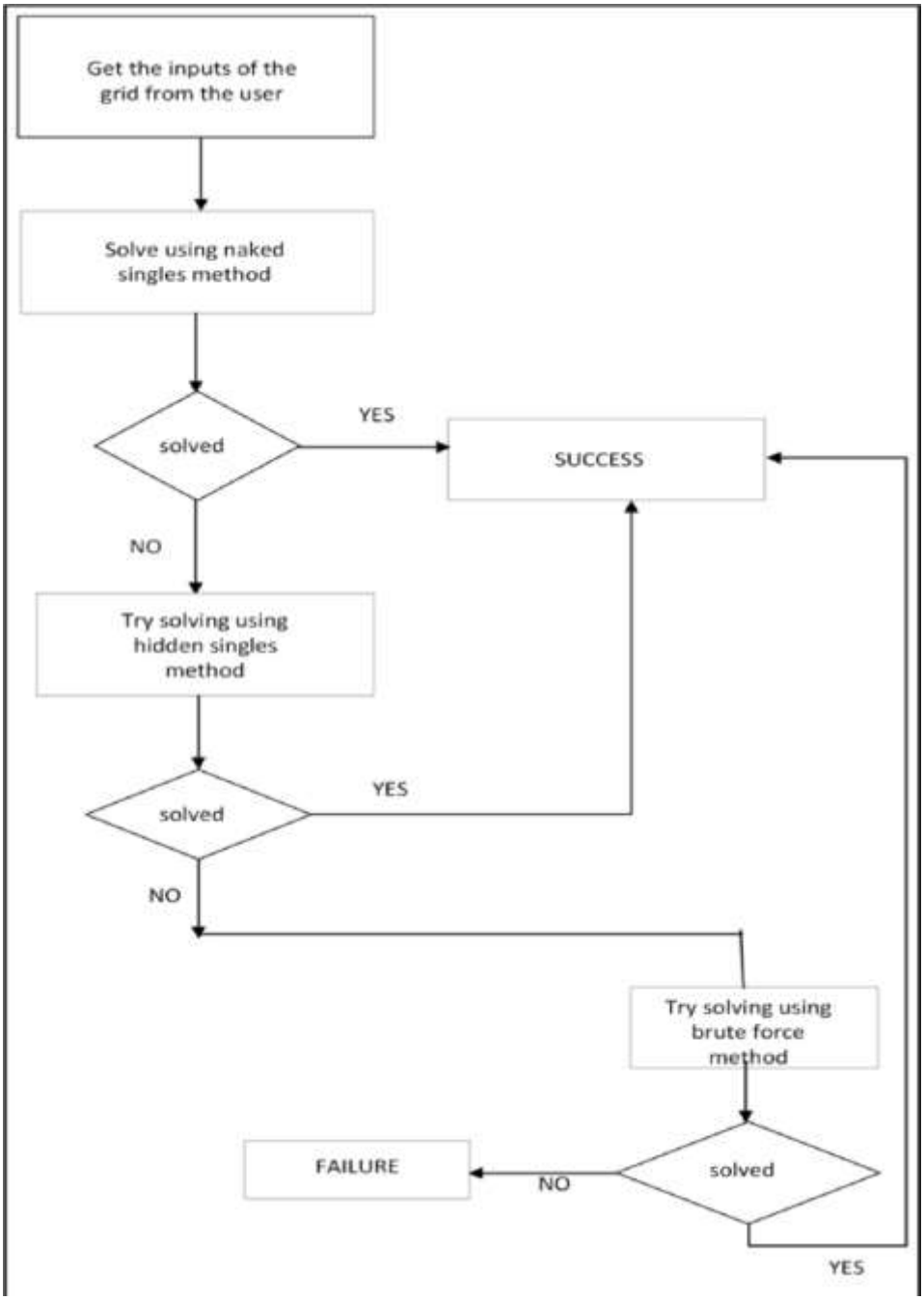
A hidden single in Sudoku is the sole number that can only be put in a specific area to complete the number combinations in its row, column, and inner box.

BRUTE FORCE METHOD:

A brute force algorithm visits the empty cells in some order, filling in digits sequentially,backtracking when the number is found to be not valid.

ARCHITECTURE DIAGRAM:

FLOW DIAGRAM:



OUTCOMES AND DELIVERABLES:

By analysing the grid with the given set of inputs, the appropriate numbers for the vacant squares of the grid are computed using the programming language C and data structure which not only helps the people to solve it quicker and with least mental effort.