**PJ 13 Report Your Name: Francisco Valadez**

**A. The following is my Java program:**

**// Please copy your Java program into here from your Eclipse window. The code must be colored.**

**// You must not copy Java program from your .java file since the code over there is not colored at all.**

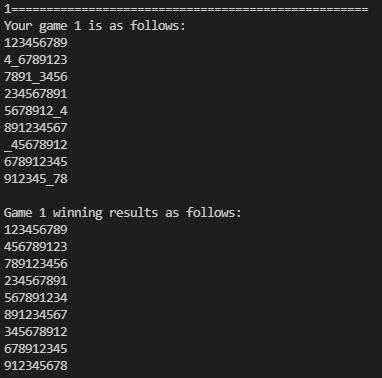
**// You must not show screen prints here.**

// Author: Francisco Valadez  
// Date: 6/7/2021  
// Purpose: This program solves 4 preloaded sudoku games!  
  
public class sudokuSolver  
{  
 //Below are the 4 preloaded games, elements with -1 are placeholders for the missing number  
 public static int G1[][] =  
 {  
 {1, 2, 3, 4, 5, 6, 7, 8, 9},  
 {4, -1, 6, 7, 8, 9, 1, 2, 3},  
 {7, 8, 9, 1, -1, 3, 4, 5, 6},  
 {2, 3, 4, 5, 6, 7, 8, 9, 1},  
 {5, 6, 7, 8, 9, 1, 2, -1, 4},  
 {8, 9, 1, 2, 3, 4, 5, 6, 7},  
 {-1, 4, 5, 6, 7, 8, 9, 1, 2},  
 {6, 7, 8, 9, 1, 2, 3, 4, 5},  
 {9, 1, 2, 3, 4, 5, -1, 7, 8},  
 };  
  
 public static int G2[][] =  
 {  
 {-1, 2, 3, 4, 5, 6, 7, 8, -1},  
 {-1, 5, 6, 7, 8, 9, -1, 2, 3},  
 {7, 8, 9, 1, -1, 3, 4, 5, 6},  
 {2, 3, -1, 5, 6, 7, 8, 9, 1},  
 {5, 6, 7, 8, 9, -1, 2, 3, 4},  
 {8, 9, -1, 2, 3, 4, 5, 6, -1},  
 {3, 4, 5, 6, 7, 8, -1, 1, 2},  
 {-1, 7, 8, 9, 1, -1, 3, 4, 5},  
 {9, 1, -1, 3, 4, 5, 6, 7, -1}  
 };  
  
 public static int G3[][] =  
 {  
 {1, 2, -1, 4, 5, 6, -1, 8, 2},  
 {-1, 5, 6, 7, 8, -1, 1 ,2 ,3},  
 {7, -1, 9, 1, 2, -1, 4, 5, 6},  
 {-1, 3, 4, -1, 6, -1, 8, 9, 1},  
 {5, 6, 7, -1, 9, 1, 2, -1, 4},  
 {8, -1, 1, 2, 3, 4, 5, -1, 7},  
 {3, 4, 5, -1, 7, 8, -1, 1 ,2},  
 {6, 7, -1, 9, 1, 2, 3, -1, 5},  
 {9, -1, 2, 3, 4, -1, 6, 7, -1}  
 };  
  
 public static int G4[][] =  
 {  
 {1, -1, 3, 4, -1, 6, 7, -1, 9},  
 {-1, 5, -1, 7, 8, -1, 1, -1, 3},  
 {-1, 8, -1, 1, 2, -1, 4, 5, -1},  
 {2, -1, 4, 5, -1, 7, 8, -1, 1},  
 {-1, 6, 7, -1, 9, 1, -1, 3, -1},  
 {8, -1, 1, -1, 3, -1, 5, 6, -1},  
 {-1, 4, 5, -1, 7, -1, 9, -1 ,2},  
 {6, -1, 8, -1, 1, -1, 3, 4, -1},  
 {-1, 1, 2, -1, 4, 5, 6, -1, 8}  
 };  
  
 //This function prints the array  
 public static void print(int[][] array)  
 {  
 for (int x = 0; x < 9; x++)  
 {  
 for (int y = 0; y < 9; y++)  
 {  
 if(array[x][y] == -1)  
 System.out.print("\_");  
 else  
 System.out.print(array[x][y]);  
 }  
 System.out.println();  
 }  
 }  
  
 //This function sends the board to be solved  
 public static void check(int[][] game, int number)  
 {  
 int[][] fixed = game;  
 if (solvegame(fixed))  
 {  
 System.out.println("\nGame " + number + " winning results as follows: ");  
 print(fixed);  
 }  
 else  
 System.out.println("Unsolvable");  
  
 System.out.println();  
 }  
  
 //This function goes through each number inthe preloaded game to find the missing numbers  
 private static boolean solvegame(int[][] game)  
 {  
  
 for (int row = 0; row < 9; row++)  
 {  
 for (int column = 0; column < 9; column++)  
 {  
 if (game[row][column] == -1)  
 {  
 for (int testNum = 1; testNum <= 9; testNum++)  
 {  
 if (isValidPlacement(game, testNum, row, column))  
 {  
 game[row][column] = testNum;  
  
 if (solvegame(game))  
 {  
 return true;  
 }  
 else  
 game[row][column] = -1;  
 }  
 }  
 return false;  
 }  
 }  
 }  
 return true;  
 }  
  
 private static boolean isValidPlacement(int[][] game, int number, int row, int column)  
 {  
 return !checkRow(game, number, row) && !checkColumn(game, number, column) && !squareCheck(game, number, row, column);  
 }  
  
 //This function checks the square for the number  
 private static boolean squareCheck(int[][] game, int number, int row, int column)  
 {  
 int squarRow = row - row % 3;  
 int squareColumn = column - column % 3;  
  
 for (int i = squarRow; i < squarRow + 3; i++)  
 {  
 for (int x = squareColumn; x < squareColumn + 3; x++)  
 {  
 if (game[i][x] == number)  
 return true;  
 }  
 }  
 return false;  
 }  
  
 //This function checks if the number is already in the column  
 private static boolean checkColumn(int[][] game, int number, int column)  
 {  
 for (int i = 0; i < 9; i++)  
 {  
 if (game[i][column] == number)  
 return true;  
 }  
 return false;  
 }  
  
 //This function checks if the number is already in the row  
 private static boolean checkRow(int[][] game, int number, int row)  
 {  
 for (int i = 0; i < 9; i++)  
 {  
 if (game[row][i] == number)  
 return true;  
 }  
 return false;  
 }  
  
 public static void main(String[] args)  
 {  
 System.out.println("Welcome to play this Sudoku real game designed by Francisco Valadez!");  
 System.out.println("1===================================================");  
 System.out.println("Your game 1 is as follows:");  
 print(G1);  
 check(G1, 1);  
 System.out.println("2===================================================");  
 System.out.println("Your game 2 is as follows:");  
 print(G2);  
 check(G2, 2);  
 System.out.println("3===================================================");  
 System.out.println("Your game 3 is as follows:");  
 print(G3);  
 check(G3, 3);  
 System.out.println("4===================================================");  
 System.out.println("Your game 4 is as follows:");  
 print(G4);  
 check(G4, 4);  
 System.out.println("5===================================================");  
 System.out.println("Thank you for playing this Sudoku real game of Francisco Valadez!");  
 System.out.println("6===================================================");  
 }  
}

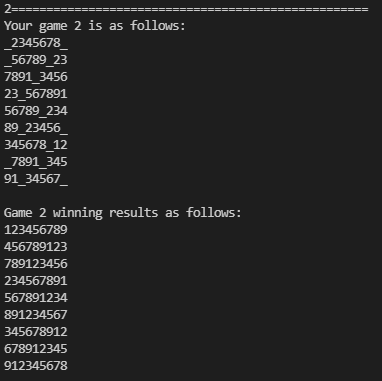
**B. The following is the complete output of my 4 games as 4 test cases:**

**// Please copy your Eclipse console output into here.**

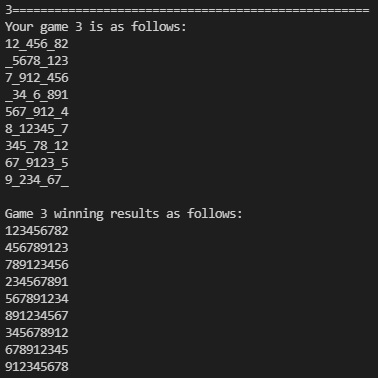
**Test Case 1:**



**Test Case 2:**



**Test Case 3:**



**Test Case 4:**

