bool isValidSudoku(char\*\* board, int boardSize, int\* boardColSize){

    bool ans=true;

        int checkaArray1[10]={0};

        int checkaArray2[10]={0};

        int checkaArray3[10]={0};

    for(int i=0;i<boardSize;i++){

        int checkRaw[10]={0};//將raw確認矩陣歸零

        if(i==0 || i==3 || i==6 ){//將確認矩陣歸零

            for(int j=0;j<10;j++){

                checkaArray1[j]=0;

                checkaArray2[j]=0;

                checkaArray3[j]=0;

            }

        }

        for(int j=0;j<boardSize;j++){

            //printf("%d ",board[i][j]-'0');

            if( (board[i][j]-'0') > 0){//計算確認矩陣中的數字

                checkRaw[board[i][j]-'0']++;

                if(  j >= 0 && j < 3){

                    checkaArray1[board[i][j]-'0']++;

                }else if( j >= 3 && j < 6){

                    checkaArray2[board[i][j]-'0']++;

                }else{

                    checkaArray3[board[i][j]-'0']++;

                }

            }

        }

        //printf("check=");

        for(int j=0;j<boardSize+1;j++){//確認確認矩陣中的數字是否有重複

            //printf("%d ",checkRaw[j]);

            if( checkRaw[j] >=2 || checkaArray1[j] >=2  ||checkaArray2[j] >=2  || checkaArray3[j] >=2 ){

                return false;

            }

        }

        //printf("\n");

    }

   //printf("\n");

    for(int i=0;i<boardSize;i++){

        int checkCol[10]={0};//將確認矩checkCol陣歸零

        for(int j=0;j<boardSize;j++){//計算確認矩陣checkCol中的數字

            //printf("%d ",board[j][i]-'0');

            if( board[j][i]-'0' > 0){

                checkCol[board[j][i]-'0']++;

            }

        }

        for(int j=0;j<boardSize+1;j++){//確認確認矩陣中的數字是否有重複

            if( checkCol[j] >=2 ){

                return false;

            }

        }

        //printf("\n");

    }

    return true;

}