```c
bool _exist(char** board, int boardSize, int* boardColSize, char* word, bool** bitmap, int x, int y, int count) {
    bool result;

    result = false;


    printf("%d, %d, %d\n", x, y, count);


    if(word[count] == '\0')
    {
        return true;
    }else
    {
        if( x < boardSize || y < boardColSize[0] )
        {
            if(word[count] == board[x][y])
            {
                bitmap[x][y] = true;
                count++;
            }

            if( (x < (boardSize - 1)) &&
                (false == bitmap[x+1][y])
              )
            {
                result = _exist(board, boardSize, boardColSize, word, bitmap, x+1, y, count);
            }

            if( ( false == result ) &&
                (x > 0) &&
                (false == bitmap[x-1][y]) &&
                ( word[count] == board[x-1][y] )
              )
            {
                result = _exist(board, boardSize, boardColSize, word,bitmap,  x-1, y, count);
            }

            if( ( false == result )  &&
                ( y < (boardColSize[0]-1) ) &&
                ( false == bitmap[x][y+1] )
              )
            {
                result = _exist(board, boardSize, boardColSize, word,bitmap,  x, y+1, count);
            }

            if( ( false == result ) &&
                ( y > 0 )  &&
                ( false == bitmap[x][y-1] ) &&
                ( word[count] == board[x][y-1] )
              )
            {
                result = _exist(board, boardSize, boardColSize, word,bitmap,  x, y-1, count);
            }

            if(word[count] == board[x][y])
            {
                bitmap[x][y] = false;
                count--;
            }
        }

        return result;
    }
}

bool dfs(char** board, int boardSize, int* boardColSize, char* word, int row, int col, int word_index)
{
    char tmp;
    if (word[word_index+1] == '\0') return true;


    if(row < (boardSize -1))
    {
        if(board[row+1][col] == word[word_index+1])
        {
            tmp = board[row+1][col];
            board[row+1][col] = '#';
            if(dfs(board, boardSize, boardColSize, word, row+1, col, word_index+1))
            {
                return true;
            }
            board[row+1][col] = tmp;
        }
    }

    if(col < (boardColSize[row]-1) )
    {
        if(board[row][col+1] == word[word_index+1])
        {
            tmp = board[row][col+1];
            board[row][col+1] = '#';
            if(dfs(board, boardSize, boardColSize, word, row, col+1, word_index+1))
            {
                return true;
            }
            board[row][col+1] = tmp;
        }
    }

    if(row > 0)
    {
        if((board[row-1][col] == word[word_index+1]))
        {
            tmp = board[row-1][col];
            board[row-1][col] = '#';
            if(dfs(board, boardSize, boardColSize, word, row-1, col, word_index+1))
            {
                return true;
            }
            board[row-1][col] = tmp;
        }
    }

    if(col > 0)
    {
        if(board[row][col-1] == word[word_index+1])
        {
            tmp = board[row][col-1];
            board[row][col-1] = '#';
            if(dfs(board, boardSize, boardColSize, word, row, col-1, word_index+1))
            {
                return true;
            }
            board[row][col-1] = tmp;
        }
    }

    return false;
}

bool exist(char** board, int boardSize, int* boardColSize, char* word) {

    int row;
    int col;
    char tmp;

    if(word == '\0')
    {
        return true;
    }


    for (row = 0; row < boardSize; row++)
    {
        for (col = 0; col < boardColSize[row]; col++)
        {
          if (board[row][col] == word[0])
          {
              tmp =  board[row][col];
                board[row][col] = '#';
            if (dfs(board, boardSize, boardColSize, word, row, col, 0))
            {
                return true;
            }
            board[row][col] = tmp;
          }
        }
    }

    return false;
}
```