



**UNIVERSIDADE FEDERAL DO CARIRI
CENTRO DE CIÊNCIAS E TECNOLOGIA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

DISCIPLINA: ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES

DOCENTE: RAMON SANTOS NEPOMUCENO

PERÍODO LETIVO: 2023.2

**ANDRÉ LEITE GONDIM
IVNA FEITOSA ALENCAR**

MINESWEEPER

**JUAZEIRO DO NORTE
2024**

O "Minesweeper" (campo minado em português) é um jogo de tabuleiro que desafia os jogadores a descobrir células sem bombas em um campo minado. Os números revelados indicam quantas bombas estão adjacentes a uma determinada célula. O objetivo é descobrir todas as células seguras sem acionar uma bomba.

Funções:

- Main.asm
- Macro.asm
- initializeBoard.asm
- plantBombs.asm
- printboard.asm
- revealNeighBoring.asm
- play.asm
- checkVictory.asm
- countAdjacentBombs.asm

Play.asm:

Essa é a função responsável por dar início ao jogo, Primeiro ele recebe as coordenadas passadas pelo usuário e verifica se não é uma bomba, caso seja o programa é encerrado e o jogo acaba, caso contrário ele chama a função para olhar quantas bombas tocam a casa escolhida pelo usuário.

OBS: as funcionalidades linha a linha estão comentadas no código.

CheckVictory.asm:

Essa função verifica se todas as células do tabuleiro, exceto as bombas, foram descobertas, indicando a vitória do jogador. Ele percorre todas as células do tabuleiro, contando quantas foram descobertas. Após verificar todas as células, ele compara o número total de células menos o número de bombas com o número de células descobertas. Se forem iguais ou maiores, o jogador vence e a função retorna 1, caso contrário retorna 0.

OBS: as funcionalidades linha a linha estão comentadas no código.

CountAdjacentBombs.asm:

A countAdjacentBombs é responsável por contar o número adjacente de bombas que há em uma determinada célula, mas sem revelar a posição de todas as bombas do tabuleiro, o que ajuda na tomada de decisões por parte do usuário, referente qual campo evitar ou escolher para revelar. Essa função é chamada pela play e RevealNeighBoring e recebe os parâmetros linha, coluna e o próprio tabuleiro. Com isso, por meio do looping percorre-se uma matriz 3x3 centrada na célula selecionada pelo jogador. Avalia-se, então, se os limites do tabuleiro não foram ultrapassados e se não há bomba naquela posição que está sendo percorrida (`board[i][j] = -1`), se isso for verdade, incrementa a variável count. Ao fim do looping retorna-se a variável count com o número de bombas adjacentes.

RevealNeighBoring.asm:

A revealNeighBoring revela as células vazias próximas à célula escolhida. Dessa forma, assim como as funções anteriormente apresentadas, recebe os parâmetros linha, coluna e o próprio tabuleiro e é chamada pela função play. A princípio, ela realiza um looping para percorrer uma matriz 3x3 centrada na célula passada como parâmetro, em busca de verificar se essas células não ultrapassam o tabuleiro e são vazias (`board[i][j] = -2`). Caso seja verdade, atribui a quantidade de bombas adjacentes ao tabuleiro e a uma variável que será avaliada em seguida. Caso essa variável seja zero, chama-se recursivamente a função passando os parâmetros atualizados. Isso ocorre até que se encontre uma bomba adjacente.