



UNIVERSIDADE FEDERAL DO CARIRI
CENTRO DE CIÊNCIAS E TECNOLOGIAS
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

ALAN GABRIEL SILVA OLIVEIRA
PEDRO DA SILVA VIANA

PROJETO I
ARQUITETURA DE COMPUTADORES

SUMÁRIO

1 Introdução.....	2
2 Função play().....	2
2.1 Uso dos registradores.....	2
3 Função countAdjacentBombs().....	2
3.1 Uso dos registradores.....	3
4 Função revealNeighboringCells().....	3
4.1 Uso dos registradores.....	3
5 Função checkVictory().....	4
5.1 Uso dos registradores.....	4

1 Introdução

A tarefa envolve a tradução do código do jogo "Campo Minado" da linguagem C para Assembly MIPS. As funções específicas que precisam ser traduzidas incluem: ``play()``, ``countAdjacentBombs()``, ``revealNeighboringCells()``, e ``checkVictory()``. Vale ressaltar que as demais funções já foram implementadas.

2 Função `play()`

Verifica se há uma bomba na posição dada. Se sim, retorna 0 indicando o fim do jogo. Caso contrário, chama a função `countAdjacentBombs` para contar as bombas adjacentes e armazenar o valor na posição dada. Se o valor for 0, chama a função `revealNeighboringCells` para revelar as casas adjacentes. Retorna 1 para confirmar que o jogo ainda não acabou.

2.1 Uso dos registradores

```
# $s0: Recebe a posição inicial da matriz
# $s1: Recebe a coordenada 'i'
# $s2: Recebe a coordenada 'j'

# $t0: Calcula 'i * 32'
# $t1: Calcula 'j * 4'
# $t0: Obtém a coordenada relativa ao início da matriz

# $s3: Recebe a coordenada do elemento na pilha
# $s4: Recebe o elemento na posição dada

# $t2: Atribui o valor -1
# $t2: Atualiza para o valor -2

# $t4: Recebe o valor de retorno da função countAdjacentBombs
```

3 Função `countAdjacentBombs()`

Inicia a contagem 'count' com 0 e utiliza dois laços for para percorrer as posições circundantes da posição dada. Se uma bomba for encontrada em alguma dessas posições, incrementa-se 1 em 'count'. Ao final, o valor de 'count' é retornado.

3.1 Uso dos registradores

\$s0: Recebe a posição de início da matriz
 # \$s1: Recebe o valor da linha (row)
 # \$s2: Recebe o valor da coluna (column)

\$s3: 'count' é inicializado com 0
 # \$t5: 'SIZE' é armazenado para referência
 # \$s4: É atribuído -1

\$t1: 'i' recebe 'row - 1'
 # \$t2: 'i' recebe 'row + 1'

\$t3: 'j' recebe 'column - 1'
 # \$t4: 'j' recebe 'column + 1'

\$t6: Calcula 'i * 32'
 # \$t7: Calcula 'j * 4'

\$t6: Obtém a coordenada relativa ao início da matriz
 # \$t6: Obtém a coordenada do elemento na pilha

\$s5: Recebe o valor da matriz na posição dada

\$v0: Recebe 'count' para retorno

4 Função revealNeighboringCells()

Essa função atribui um valor ao espaço escolhido e, de forma recursiva, atribui valores aos espaços adjacentes se estes estiverem marcados com zero bombas ao redor. Caso contrário, a recursão é interrompida.

4.1 Uso dos registradores

\$s0: Armazena o início dos dados obtidos pela função no registrador \$a0
 # \$s1: Captura a posição x do vetor fornecido pelo registrador \$a1
 # \$s2: Obtém a posição y do vetor fornecido pelo registrador \$a2

\$s3: Representa a variável de controle do primeiro loop (for)
 # \$s4: Representa a variável de controle do segundo loop (for)
 # \$s5: Captura a posição do elemento no vetor

\$t0: Calcula \$s1 + 1
 # \$t1: Calcula \$s2 + 1
 # \$t2: Armazena a constante SIZE.
 # \$t3: Captura o valor real da coordenada x e também recupera o valor expresso no vetor

 # \$t4: Obtém o verdadeiro valor da coordenada y
 # \$t7: Armazena o valor retornado pela função countAdjacentBombs
 # \$t8: Recebe o valor -2 por atribuição direta

5 Função checkVictory()

Esta função percorre todo o vetor contando os espaços descobertos pelo jogador. Em seguida, verifica se todos os espaços foram descobertos e se alguma bomba foi ativada. Se todos os espaços foram descobertos, a função retorna 0; caso contrário, retorna 1.

5.1 Uso dos registradores

\$s0: Obtém o início dos dados pela função, utilizando o marcador \$a0.
 # \$s1: Representa a variável de controle do primeiro loop (for).
 # \$s2: Representa a variável de controle do segundo loop (for).

 # \$t0: Captura a constante SIZE.
 # \$t1: Obtém o valor real da coordenada x.
 # \$t2: Obtém o verdadeiro valor da coordenada y e o utiliza para obter a posição do elemento.

 # \$t3: Armazena a verdadeira posição do vetor.
 # \$t4: Obtém o valor do quadrado de SIZE e é posteriormente utilizado para armazenar o quadrado de SIZE menos BOMB_COUNT.
 # \$t5: Adiciona e armazena BOMB_COUNT.