TAB\_tpCondRet TAB\_DestruirTabuleiro(ptTabuleiro tabu) {

int i=0,j = 0;

while(i<8) {

j=0;

while(j<8) {

if(tabu->tab[i][j].elemento != NULL) {

LIS\_DestroiLista(tabu->tab[i][j].ameacados);

LIS\_DestroiLista(tabu->tab[i][j].ameacantes);

PEC\_LiberaPeca((Peca\*)tabu->tab[i][j].elemento);

}

j++;

}

i++;

}

LIS\_DestroiLista(listaPecas);

return TAB\_CondRetOK;

}

Argumentação de Sequência:

AE: A função de destruir tabuleiro recebe um ponteiro para a struct TAG\_tabuleiro.

AS: Ao final da função o tabuleiro foi liberado da mémoria do computador, ou seja, destruído

AI1- Declaro i e j, porém i é inicializada apontando para a primeira linha

AI2- Uso a função para destruir para destruir a lista de peças

AI3 - Significa que a função acabou sem problemas

Argumentação de Repetição

AE => AINV

-Pela a assertiva de entrada a variável i aponta para a primeira linha do tabuleiro

AE && (C == F) => AS

-Não executará a repetição, ou seja, todas as linhas e colunas foram percorridas

AE && (C == T) + B2 => AINV

-Pela AE, i aponta para a primeira linha do tabuleiro, quando for executado o bloco 2 o i será atualizado. Vale a AINV

AINV && (C == T) + B2 => AINV

-Para AINV continuar valendo o i tem que está dentro dos limites do tabuleiro

AINV && (C == F) => AS

-A variável i está fora dos limites do tabuleiro

Argumentação de Sequência

AE: j aponta para a primeira coluna e i para a linha do valor corrente

AS: j aponta para a última coluna e i para a linda do valor corrente

AI4- j aponta para a primeira coluna

AI5- i aponta para o próxima linha do tabuleiro

Argumentação de Repetição

AE => AINV

-Pela a assertiva de entrada a variável j aponta para a primeira linha do tabuleiro

AE && (C == F) => AS

-Não executará a repetição, ou seja todas as colunas foram percorridas

AE && (C == T) + B3 => AINV

-Pela AE, j aponta para a primeira coluna do tabuleiro, quando for executado o bloco 3 o j será atualizado. Vale a AINV

AINV && (C == T) + B3 => AINV

-Para AINV continuar valendo o j tem que está dentro dos limites do tabuleiro

AINV && (C == F) => AS

-A variável j está fora dos limites do tabuleiro

Argumentação de Sequência

AE: -Com a i e j definidos e verdadeiro na AINV, serviram para indicar a posição no tabuleiro a ser testada e destruída

AS: -A lista de ameaçados, ameaçantes e a peça foram destruídas

AI6- j aponta para a próxima coluna

Argumentação de Seleção

AE && (C == T) + B4 => AS

AE: AI5

AE && (C == F) => AS

-Ignorará o comando de seleção

Argumentação de Sequência

AE: -O endereço apontado no tabuleiro na linha e coluna (i, j) não são nulos

AS: -A lista de ameaçados, ameaçantes e a peça foram destruídas

AI7 - Destroi a lista de ameçados

AI8 - Destroi a lista de ameçantes

AI9 - Destroi a peça