

Argumentação de Corretude

```
/*  
*  
* Função: SEQFIM Preencher Sequencia final  
*****/
```

AE

```
SEQFIM_tpCondRet SEQFIM_PreencheSeqFim ( SEQFIM_tppSeqFim  
pSeqFim , PIL_tppPilha pPilha )  
{
```

```
    CAR_tppCarta pCartaAux;
```

```
    if( pSeqFim == NULL )  
    {  
        return SEQFIM_CondRetSeqNaoExiste ;  
    }
```

AI1

```
    while( PIL_PopCarta( pPilha, &pCartaAux ) == PIL_CondRetOK )  
    {  
        PIL_PushCarta( pSeqFim->pPilha, pCartaAux );  
    }
```

```
    return SEQFIM_CondRetOK ;
```

```
}/* Fim função: SEQFIM Preencher Sequencia final */
```

AS

AE → existe pPilha.

AS → pSeqFim foi preenchida e retorna condição de retorno CondRetOK ou é nula e retorna condição de retorno SeqNaoExiste.

Seleção:

AE → **AE** da função

AS → **AI1**

AI1 → retorna SeqNaoExiste se pSeqFim é nula. Senão, existe pPilha e pSeqFim não é nula.

1) AE && (C==T) + B → **AS**

Pela AE, a pPilha existe. Como (C==T), a sequência final não existe. Neste Caso, B retorna condição de retorno SeqNaoExiste, valendo assim a AS.

2) AE && (C==F) → **AS**

Pela AE, existe pPilha. Com (C==F), a sequência existe, valendo a AS.

Repetição:

AE → existe pPilha e pSeqFim não é nula.

AS → **AS** da função

AINV → existem 2 conjuntos:

- a retirar
- já retirados

1) AE → **AINV**

Pela AE, existe sequência e ela não é vazia. Nesse caso todos os elementos estão no conjunto a retirar, e vale AINV, pois existem 2 conjuntos.

2) AE && (C==F) → **AS**

Pela AE, como (C==F), a sequência é vazia. Vale a AS pois a sequência é nula e o retorno é SeqNaoExiste.

3) AE && (C==T) + B → **AINV**

Pela AE, como (C==T), o elemento passa para o conjunto já retirado e vale AINV.

4) $AINV \ \&\& \ (C==T) + B \rightarrow AINV$

Para que AINV continue valendo, B tem que garantir que um elemento passe do conjunto a retirar para já retirado.

5) $AINV \ \&\& \ (C==F) \rightarrow AS$

Com $(C==F)$, a sequência já foi preenchida, valendo a AS.

6) Término

O conjunto a retirar possui um número finito de elementos e a cada ciclo um deles passa para o conjunto já retirado. A repetição então termina após um número finito de passos.