

# Argumentação de Corretude

Abaixo virá escrito o código da função escolhida para a argumentação de corretude. No meio dela, em negrito e sublinhado estarão os códigos das assertivas que serão referenciados mais tarde no documento para descrever essas assertivas.

## **AE**

```
CRO_tpCondRet CRO_ImprimeListaRecurso( tcCronograma * cCronograma)
{
    tcRecurso * recursoCorrente = NULL;
    (1) if (cCronograma == NULL || cCronograma->cronograma == NULL)
        return CRO_CondRetCronogramaNaoExiste;

    AI1
    (2) if (LIS_VerificarVazia(cCronograma->cronograma->listaRecursos) ==
LIS_CondRetListaVazia)
    {
        printf("A lista de recursos está vazia\n");
        return CRO_CondRetOK;
    }
    else
    {
        printf("\nLista de Recursos do Cronograma\n");
    }

    AI2
    IrInicioLista(cCronograma->cronograma->listaRecursos);
    AI3
    recursoCorrente = (tcRecurso *)
LIS_ObterValor(cCronograma->cronograma->listaRecursos);
    REC_ImprimeRecurso(recursoCorrente);
    AI4
    (1) while
(LIS_AvançarElementoCorrente(cCronograma->cronograma->listaRecursos, 1) !=
LIS_CondRetFimLista)
    {
        printf("\n");
        recursoCorrente = (tcRecurso *)
LIS_ObterValor(cCronograma->cronograma->listaRecursos);
        REC_ImprimeRecurso(recursoCorrente);
    }
    printf("\n");
    return CRO_CondRetOK;

} /* Fim função: CRO Imprime Lista Recurso */
```

## **AS**

## AE

- A função recebe um ponteiro para uma estrutura de Cronograma
- Valem as assertivas estruturais do Cronograma
- A estrutura de cronograma recebida pela função pode estar vazia ou não

## Argumentação de Seleção do if (1)

- AE
  - Igual a AE principal
- AE && (C == T) + B => AS
  - Pela AE o ponteiro que deveria carregar a estrutura cronograma recebido pela função pode estar vazio. Como (C==T) está vazio e o código retorna a condição de retorno que descreve tal ato e encerra a função. Valendo a AS.
- AE && (C==F) => AS
  - Pela AE o ponteiro que deveria carregar a estrutura cronograma recebido pela função pode não estar vazio. Como (C==F) não está vazio e a função pode continuar tendo a certeza de que o ponteiro aponta para uma estrutura de cronograma de fato. Valendo a AS.
- AS
  - Chegando aqui ou o ponteiro do cronograma aponta para uma estrutura vazia e por isso a função encerra sua execução retornando o valor adequado e valendo as Assertivas de Saída principal ou ele aponta para uma estrutura que não está vazia e a função continua executando seus demais comandos

## AI1

- O ponteiro está apontando para uma estrutura do tipo cronograma não nula.
- A lista de recursos da estrutura cronograma pode estar vazia.

## Argumentação de Seleção do if (2)

- AE
  - Igual ao AI1
- AE && (C == T) + B1 => AS
  - Pela AE a lista de recursos da estrutura cronograma passada pode estar vazio. Como (C==T) a lista está vazia e a função imprime a mensagem “A lista de recursos está vazia” e retorna a condição de retorno adequada, valendo a AS.
- AE && (C==F) + B2 => AS
  - Pela AE a lista de recursos da estrutura cronograma passada pode estar vazio. Como (C==F) a lista não está vazia e a função imprime a mensagem “Lista de Recursos do Cronograma” valendo a AS.
- AS
  - Ou a lista está vazia, e nesse caso a tela mostra a mensagem “A lista de recursos está vazia” e a função encerra sua execução retornando o valor adequado e valendo as Assertivas de Saída principal ou a lista não está vazia, a tela apresenta a mensagem “Lista de Recursos do Cronograma” e a função continua executando seus demais comandos.

## AI2

- A mensagem “Lista de Recursos do Cronograma” está impressa na tela mas não se tem certeza de onde o ponteiro para o elemento corrente dessa lista se encontra.

## AI3

- Reposiciona-se o ponteiro para o elemento corrente da lista de recursos e o faz apontar para o primeiro elemento desta lista.

## AI4

- A variável “recursoCorrente” contém o valor do primeiro recurso da lista de recursos do cronograma.
- As informações do primeiro recurso da lista de recursos do cronograma estão impressas na tela.
- o ponteiro do elemento corrente da lista de recursos está posicionado na primeira posição da lista.

## Argumentação de Repetição do while (1)

- AE
  - Igual ao AI4
- AS
  - O ponteiro para o elemento corrente da lista de recursos do cronograma está apontando para o último elemento da lista.
  - As informações de todos os recursos contidos na lista foram impressas (com exceção da do primeiro que foi impressa antes da estrutura de repetição).
- AINV
  - Existem dois conjuntos: a imprimir e já impressos.
  - O ponteiro para o elemento corrente da lista de recursos aponta para o último elemento do conjunto já impresso.
- $AE \Rightarrow AINV$ 
  - Pela AE, o ponteiro para o elemento corrente da lista de recursos aponta para o primeiro elemento da lista. Somente o primeiro elemento da lista já foi impresso na tela e está no conjuntos dos elementos já impressos, os demais estão no conjunto a imprimir. Vale a Assertiva Invariante.
- $AE \ \&\& \ (C==F) \Rightarrow AS$ 
  - Caso a lista só possua um elemento, quando a função tentar mover o ponteiro para o elemento corrente da lista de recursos para o próximo elemento, pela lista não possuir um elemento seguinte, a função para mover o ponteiro retornará com a condição de que já alcançou o final da lista. Como  $(C==T)$  sabemos que a lista chegou ao fim. Vale a AS pois não tem mais elementos para imprimir.
- $AE \ \&\& \ (C==T) + B \Rightarrow AINV$ 
  - Pela AE o ponteiro para o elemento corrente da lista de recursos do cronograma aponta para um elemento já impresso. Como  $(C==T)$  a lista conseguiu mover seu ponteiro corrente para um próximo elemento que não foi impresso ainda. Esse mesmo será impresso, passando do conjunto a imprimir para o já impresso. Assim o ponteiro volta a apontar para o último elemento do conjunto já impresso, valendo AINV.
- $AINV \ \&\& \ (C==T) + B \Rightarrow AINV$ 
  - Como  $(C==T)$  o ponteiro para o elemento corrente da lista de recursos do cronograma é reposicionado para um elemento que ainda precisa ser

impresso e o mesmo é impresso, passando do conjunto dos elementos a serem impressos para o conjunto dos elementos já impressos. Ao final de cada ciclo vale AINV.

- AINV && (C==F) => AS
  - Para (C==F) a lista chegou ao seu fim e portanto não tem mais elementos para serem impressos, valendo a AS.
- Término
  - Como a cada ciclo um elemento é retirado do conjunto a imprimir e este possui um número finito de elementos, a repetição termina em um número finito de passos.

## AS

- Ainda valem as assertivas estruturais do Cronograma.
- É apresentado na tela ou nenhuma mensagem, ou “A lista de recursos está vazia” ou a mensagem “Lista de Recursos do Cronograma” seguida das informações de cada um dos recursos contidos na lista.