

USP – ICMC – SSC

**SSC0603 – Estrutura de Dados I (ED1) – 2025/2**  
**TRABALHO TP01 – ACME Ranking List Manager**

**ACME Ranking List Manager (RLM)** – Trabalho INDIVIDUAL

\*\*\*\*\*

**Versão 1.0r00** - VER sempre o PDF com descrição completa!

**ATENÇÃO: DATA LIMITE DE UPLOAD 19/10/25 23:55:55**

**(Sistema fecha automaticamente a submissão na data/hora marcada no RunCodes)**

> Use listas **LDED Lista Dinâmica Encadeada Dupla** neste trabalho[TAD Backes, FOsorio, ou outro]

> Criar um programa que gerencia listas de “**Rankings**” (**RLM**) onde o usuário define qual a ordem dos elementos ranqueados. Exemplos de Aplicações;

- Organizar o Ranking (ordem de classificação) dos carros de Formula 1 (F1), durante a corrida (colocação parcial e final) de cada carro/piloto/equipe;
- Organizar o Ranking de qual foi o Coiote que chegou mais perto de pegar o Papa-Léguas, usando os produtos ACME;
- Prestar serviços para a Máfia Italiana, Japonesa ou do Trump, criando a lista do “chefe”, e ordenando seus “prestadores de serviço” de acordo com o “valor” estimado de cada um;

Note que não existe uma “ordem a priori”, sendo o dono da lista que define esta ordem, de acordo com critérios próprios seus. O ranking é o ordenado do mais importante (início da lista) para o menos importante (menos significativo), no final da lista. Como toda lista de Ranking “não formal”, novos elementos podem ser inseridos na lista (em qualquer posição), ou removidos (eliminados).

Os dados inseridos são: ID (Nro. Identificador Privativo do Indivíduo), Nome (Nome ou Apelido). O ID é um número inteiro e o Nome é uma palavra de no máximo 15 caracteres (sem espaços).

As principais funções disponíveis são as indicadas abaixo, e incluem:

- (i) Inserção no Início, (ii) Inserção no Final, (iii) Inserir Depois/Antes do Atual (cursor do Buscar ID),
- (iv) Buscar ID (move o cursor), (v) Remover item buscado (ID, cursor atual move p/anterior),
- (vi) Move o Cursor para o Início, (vii) Move o Cursor para o Fim,
- (viii) Listar Todos, (ix) Listar vizinhança do cursor: Anterior+Atual+Próximo, (x) Exibir Atual (cursor).

O cursor é um ponteiro que vai “passear” na lista, e que aponta para o nodo de interesse atual.

> Entrada: o usuário digita uma **sequência de comandos para o Gerenciador de Rankings (RLM)**, como listado a seguir. Os comandos são SEMPRE neste formato:

<Cmd><espaço><ID><espaço><NOME><enter> ou

<Cmd><espaço><ID><enter> ou

<Cmd><enter>

Onde:

<Cmd> - Uma única letra em maiúscula, que indica qual comando executar

<ID> - Número inteiro que identifica o nodo de forma única (não duplicado).

<NOME> - Palavra de até 15 caracteres (sem espaços, sem acentos)

A lista de comandos sempre termina pelo comando: "X"" (eXit – Termina a execução do RLM). Se houver um comando inválido, ignorar. Se o comando não puder ser executado, ignorar. É o usuário que garante que não haverá IDs duplicados na inserção (não precisa verificar).

## >> Comandos do Editor

<Cmd><espaço><ID><espaço><NOME><enter>

```
I <ID> <NOME>      // Insere um nodo no inicio (I) da lista (inicio do Ranking List)
F <ID> <NOME>      // Insere um nodo no final (F) da lista (final do Ranking List)
A <ID> <NOME>      // Insere um nodo antes (A) do atual nodo apontado na lista
                     // (insere antes do cursor, cursor aponta para o nodo inserido)
D <ID> <NOME>      // Insere um nodo depois(D) do atual nodo apontada na lista
                     // (insere depois do cursor, cursor aponta para o nodo inserido)
```

<Cmd><espaço><ID><enter>

```
B <ID>              // Procura um ID a partir do inicio da lista,
                     // e colocar o cursor no nodo encontrado.
                     // Se não achar o ID, mantém o cursor onde está
                     // (não move o cursor se não achou o ID procurado)
                     // Se quiser procurar em toda lista, basta dar um:
                     // "S" (Ranking Start) e depois "B <ID>"
                     // Se quiser procurar e exibir a próxima ocorrência do ID:
                     // "B" <ID>" (busca), "N" (next), "P" (print atual)
```

<Cmd><enter>

```
R                  // Remove o nodo atual que está sendo apontado pelo cursor
                     // Depois de remover o cursor vai para o nodo imediatamente anterior.

S                  // Posiciona o cursor no inicio da lista (S = "Start").
E                  // Posiciona o cursor no final da lista (E = "End").
N                  // Posiciona o cursor no nodo seguinte (N = "Next").

P                  // Exibe somente o nodo atual onde está o cursor (P = "Print").
L                  // Lista todos os nodos do inicio ao fim
V                  // Exibe os vizinhos, ou seja, nodo anterior, atual (cursor) e próximo

X                  // Termina a execução e sai do programa
                     // A lista de comandos SEMPRE termina por um comando X.
                     // Se não tivesse o X, o programa ficaria em LOOP esperando comandos.
```

Exemplos de Uso (casos de teste):

Entrada:	Saída:	Caso 1
I 10 Dez	1 Um	
I 1 Um	10 Dez	
L		
X		

Entrada:	Saída:	Caso 2
I 10 Dez	10 Dez	
F 1 Ultimo	5 Meio	
B 10	1 Ultimo	
D 5 Meio		
B 5		
V		
X		

Entrada:	Saída:	Caso 3
I 1 Um	0 Chefe	
F 2 Dois	1 Um	
F 3 Tres	7 Furafila	
F 4 Quatro	2 Dois	
F 5 Cinco	3 Tres	
I 0 Chefe	4 Quatro	
B 2	5 Cinco	
A 7 Furafila		
L		
X		

Entrada:	Saída:	Caso 4
I 1 Um	0 Chefe	
F 2 Dois		
F 3 Tres		
F 4 Quatro		
F 5 Cinco		
I 0 Chefe		
S		
B 99		
P		
X		

Entrada:	Saída:	Caso 5
I 1 Um	0 Chefe	
F 2 Dois	1 Um	
F 3 Tres	2 Dois	
F 4 Quatro	4 Quatro	
F 5 Cinco	5 Cinco	
I 0 Chefe		
B 3		
R		
L		
X		

=====

Resumo dos Comandos:

I - Insere Início ID Nome  
F - Insere Fim ID Nome  
A - Insere Antes ID Nome  
D - Insere Depois ID Nome  
B - Busca ID  
R - Remove atual (cursor)  
S - Move cursor para o início (start)  
E - Move cursor para o fim (end)  
N - Move cursor para o próximo (next)  
P - Exibe (print) atual (cursor)  
L - Exibe toda Lista  
V - Exibe vizinhos (anterior+atual+proximo)  
X - Fim do Programa.

=====  
F.Osório  
Set.2025  
=====