**Lista Simplesmente Encadeada Dinâmica**

|  |
| --- |
| **ALGORITMO .::. CriarLSE** |
| *Entrada*: --  *Saída*: Ptlista (Tipo Nó); |
| **Início**  PtLista ← nulo  **fim** |

|  |
| --- |
| **ALGORITMO .::. inserirInicioLSE** |
| *Entrada*: PtLista (Tipo Nó); Dado (Tipo dado)  *Saída*: PtLista (Tipo Nó); sucesso (lógico)  Variável auxiliar: Novo (Tipo Nó) |
| Início  alocar(Novo);  Se Novo = nulo **então**  sucesso←falso  **Senão**  **inicio**  Novo↑.info ← dado  Novo↑.prox ← PtLista  PtLista ← Novo  sucesso ← verdadeiro  **fim**  **fim** |

|  |
| --- |
| **ALGORITMO .::. inserirFimLSE** |
| *Entrada*: PtLista (Tipo Nó); Dado (Tipo dado)  *Saída*: PtLista (Tipo Nó); sucesso (lógico)  Variáveis auxiliares: Novo, p (Tipo Nó) |
| **Início**  alocar(Novo);  **Se** Novo = nulo **então**  sucesso←falso  **Senão**  **inicio**  Novo↑.info ← dado  Novo↑.prox ← nulo  **Se** PtLista = nulo **então**  PtLista ← Novo  **Senão**  **Início**  p ← PtLista  **Enquanto** p↑.prox ≠ nulo **Faça**  **inicio**  p ← p↑.prox  **fim**  p↑.prox ← Novo  **fim**  sucesso ← verdadeiro  **fim** |

|  |
| --- |
| **ALGORITMO .::. numeroDeElementosLSE** |
| *Entrada*: PtLista (Tipo Nó)  *Saída*: N (Inteiro)  Variável auxiliar: Pt(Tipo Nó) |
| **Início**  N ← 0  Pt ← PtLista  **Enquanto** Pt ≠ nulo **faça**  **Inicio**  N ← N + 1  Pt ← Pt↑.prox  **fim**  **fim** |

|  |
| --- |
| **ALGORITMO .::. inserirPosicaoLSE** |
| *Entrada*: PtLista (Tipo Nó); k (Tipo inteiro); Dado (Tipo dado)  *Saída*: PtLista (Tipo Nó); sucesso (lógico)  Variável auxiliar: Novo, PtAnt (Tipo Nó), N, i (inteiro) |
| **Início**  N ← numeroDeElementos()  **Se ((**k < 1 ) ou (k > N + 1)) **então**  sucesso ← falso  **Senão se** k = 1  **Então**  **inicio**  {INSERIR INICIO}  Sucesso ← verdadeiro  **Fim**  **Senão**  **Inicio**  Alocar(novo)  **Se** Novo = nulo **então**  **Inicio**  Sucesso ← falso  **Fim**  **Senão**  **Inicio**  PtAnt ← PtLista  **Para** i **de** 1 **até** k – 2 **faça**  **Inicio**  PtAnt ← PtAnt↑.prox  **Fim**  Novo↑.info ← dado  Novo↑.prox ← PtAnt↑.prox  PtAnt↑.prox ← Novo  Sucesso ← verdadeiro  **fim**  **Fim**  **Fim** |

|  |
| --- |
| **ALGORITMO .::. excluirInicioLSE** |
| *Entrada*: PtLista (Tipo Nó)  *Saída*: PtLista (Tipo Nó); sucesso (lógico)  Variável auxiliar: P (Tipo Nó) |
| **Início**  **Se** PtLista = nulo então  Sucesso ← falso  **Senão**  **inicio**  P ←PtLista  PtLista ← PtLista↑.prox  Liberar(P)  sucesso ← verdadeiro  **fim**  **fim** |

|  |
| --- |
| **ALGORITMO .::. excluirInFimLSE** |
| *Entrada*: PtLista (Tipo Nó)  *Saída*: PtLista (Tipo Nó); sucesso (lógico)  Variável auxiliar: P, Q (Tipo Nó) |
| **Início**  **Se** PtLista ≠ nulo **então**  **Inicio**  **Se** PtLista↑.prox = nulo **então**  **Inicio**  {EXCLUIRINICIO}  Sucesso ← verdadeiro  **Fim**  **Senão**  **Inicio**  P ←PtLista  **Enquanto** P↑.prox↑.prox ≠ nulo **faça**  **Inicio**  P ← P↑.prox  **Fim**  Q ← P↑.prox  P↑.prox ← nulo  Liberar(Q)  Sucesso ← verdadeiro  **fim**  **fim**  **senão**  **inicio**  sucesso← falso  **fim**  **fim** |

|  |
| --- |
| **ALGORITMO .::. excluirPosicaoLSE** |
| *Entrada*: PtLista (Tipo Nó); k (inteiro)  *Saída*: PtLista (Tipo Nó); sucesso (lógico)  Variáveis auxiliares: P , Q , PtAnt(Tipo Nó); i, N (inteiro) |
| **Início**  N ← numeroDeElementosLSE()  **Se** ((PtLista = nulo) ou ((k < 1) ou (k > N ))) **então**  **inicio**  sucesso←falso  **fim**  **Senão se**  k = 1 **então**  **inicio**  {EXCLUIR INICIO}  Sucesso ← verdadeiro  **Fim**  **Senão**  **Inicio**  P ← PtLista  **Para** i de 1 ate k – 1 **faça**  **Inicio**  PtAnt ← P;  P ← P↑.prox  **Fim**  PtAnt↑.prox ← P↑.prox  Desalocar(P)  Sucesso ← verdadeiro  **Fim**  **Fim** |

|  |
| --- |
| **ALGORITMO .::. PesquisarPosicaoLSE** |
| *Entrada*: PtLista (Tipo Nó), pos(inteiro)  *Saída*: PtLista (Tipo Nó)  Variável auxiliar: item (Tipo Nó), i, N (inteiro) |
| **Início**  N ← NumeroDeElementos()  **Se** PtLista = nulo **então**  **inicio**  Escrever(Lista vazia)  Item ← nulo  **Fim**  **Senão se** (pos < 1) ou (pos > N) **então**  **Inicio**  Escrever(Posicao inválida)  Item ← nulo  **Fim**  **Senão**  **inicio**  item ← PtLista  **Para** i de 1 **até** pos – 1 **faça**  Item ← item↑.prox  **Fim**  **Fim** |

|  |
| --- |
| **ALGORITMO .::. PesquisarPorElementoLSE** |
| *Entrada*: PtLista (Tipo Nó), Dado(Tipo dado)  *Saída*: item (Tipo Nó), |
| **Início**  **Se** PtLista = nulo **então**  **inicio**  Escrever(Lista vazia)  Item ← nulo  **Fim**  **Senão**  **inicio**  item ← PtLista  **Enquanto** item ≠ nulo **faça**  **Inicio**  **Se** item↑.info = Dado **então**  **Retorne** item  Item ← item↑.prox  **Fim**  item ← nulo  **Fim**  **Fim** |