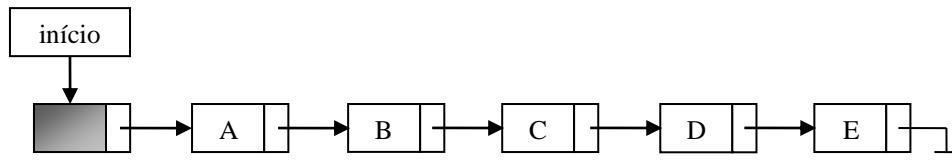


### 3ª Lista de exercícios

---

3.1 Às vezes, uma pequena modificação na estrutura de dados utilizada provoca melhorias nos algoritmos de manipulação da estrutura de dados. Considere uma lista encadeada na qual o primeiro nó é sempre ignorado para armazenamento de informações, ou seja, as informações são sempre alocadas a partir do segundo nó. Esquematicamente, a lista fica organizada como na figura abaixo:



Crie uma aplicação para manipular uma lista encadeada com informações sendo alocadas a partir do segundo nó. Implemente funções para inserção e remoção de um nó, impressão e ordenação da lista.

3.2 Reescreva o exercício anterior transformando a variável “início” de global para local ao módulo principal de seu programa.

3.3 Implemente uma pilha com alocação encadeada, inserindo dados sempre na extremidade oposta ao nó inicial. Escreva módulos para inclusão e exclusão de um nó e para impressão da pilha.

3.4 Reescreva o exercício anterior inserindo dados na pilha sempre na mesma extremidade do nó inicial.

3.5 Implemente uma fila com alocação encadeada, inserindo dados sempre na extremidade oposta ao nó inicial. Escreva módulos para inclusão e exclusão de um nó e para impressão da fila.

3.6 Reescreva o exercício anterior inserindo dados na fila sempre na mesma extremidade do nó inicial.

3.7 Seja L uma lista simplesmente encadeada, composta dos números  $l_1, l_2, l_3, \dots, l_n$ , respectivamente, segundo a ordem de armazenamento. Escreva um algoritmo que, percorrendo L uma única vez, constrói outra lista L', formada dos seguintes elementos.

a)  $l_2, l_3, \dots, l_n, l_1$

b)  $l_n, \dots, l_3, l_2, l_1$