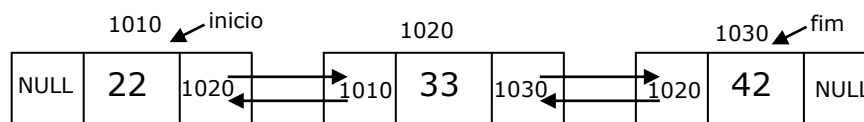


Aula 6 – Listas Duplamente Encadeadas

Uma lista duplamente encadeada é aquela em que cada nodo possui duas referências ao invés de uma apenas, como é a Lista Simplesmente Encadeada, permitindo assim que a lista seja percorrida nos dois sentidos – do início para o fim e do fim para o início. Para que isso seja possível cada nodo apresenta dois campos de ele (dois ponteiros), um que armazena o endereço do nodo **anterior** (*ant) e o outro que armazena o endereço do nodo seguinte, o **próximo** nodo (*prox).

O primeiro nodo da lista terá sempre o apontador para o nodo anterior no endereço nulo (NULL), e o último terá o endereço nulo (NULL) no campo que indica o próximo nodo da lista.



A implementação das operações fica facilitada quando é utilizado o duplo encadeamento, pelo fato de se poder percorrer a lista em qualquer direção, a qualquer momento. Apesar dessa forma de representação de lista apresentar um nodo mais complexo, com mais informações, a sua facilidade de manipulação justifica sua grande utilização.

- Declaração da Estrutura em C:

```
struct nodod{
    struct nodod *ant;
    int dados;
    struct nodod *prox;
};
```

- Função para a Inserção a Direita (Fim) na LDE

```
int InsereDireitaLDE(struct nodod **inicio, struct nodod **fim, int v) {
    struct nodod *novo=NULL;
    novo=(struct nodod *)malloc(sizeof(struct nodod));
    if(novo==NULL) printf("\nNao foi possivel alocar!\n");
    else{
        novo->dados=v;
        novo->prox=NULL;
        novo->ant=*fim;
        if(*inicio==NULL) {*inicio=novo;}
        else {(*fim)->prox=novo;}
        *fim=novo;
    } //fim do else
}
```