Deques

Disciplina: Estrutura de Dados

Luciano Moraes Da Luz Brum

Universidade Federal do Pampa – Unipampa – Campus Bagé

Email: lucianobrum18@gmail.com

unipampa
Universidade Federal do Pampa

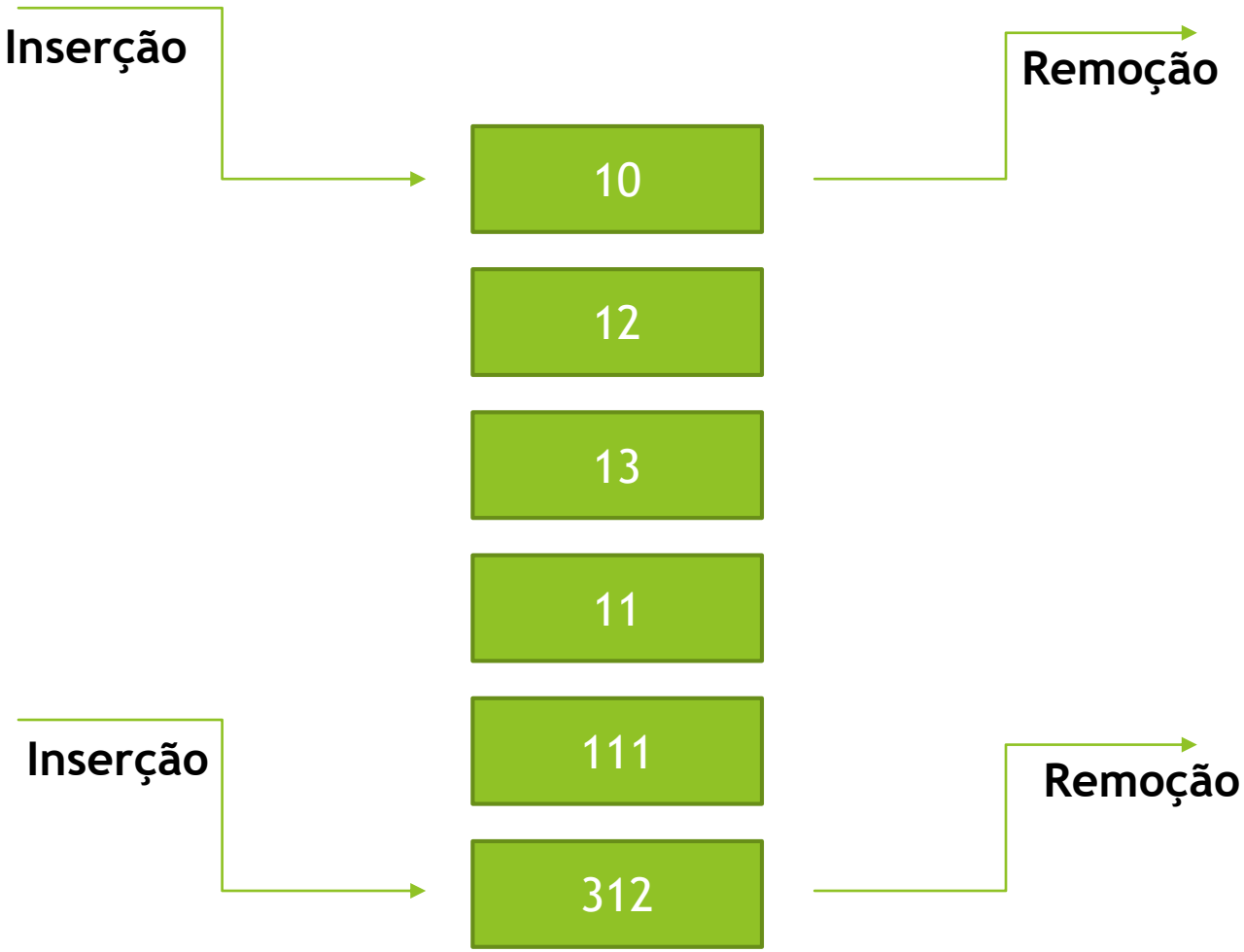
Tópicos

- O que é um deque?
- Interface do tipo Deque.
- Implementação de Deque com vetor (contiguidade física).
- Implementação de Deque com listas encadeadas (alocação dinâmica).
- Resumo.

O que é um Deque?

- Deque é uma estrutura de dados em que:
  - O elemento pode ser inserido no início ou no fim da fila. O elemento pode ser retirado também do início ou fim da fila.
- **É uma generalização das estruturas Fila e Pilha.**

O que é um Deque?



O que é um Deque?

- Ideia fundamental: a inserção de elementos será sempre no começo ou no final e a retirada dos elementos será sempre do início ou do final do deque.

- **Também conhecido como Double Ended QUEue (Fila de duas saídas ou fila dupla).**

unipampa
Universidade Federal do Pampa

Tópicos

- ~~O que é um Deque?~~
- Interface do tipo Deque.
- Implementação de Deque com vetor (contiguidade física).
- Implementação de Deque com listas encadeadas (alocação dinâmica).
- Resumo.

Interface do tipo Deque

- **Vamos considerar a implementação das seguintes operações:**
  - Criar um deque vazio;
  - Inserir elemento no início;
  - Inserir elemento no fim;
  - Remover elemento do início;
  - Remover elemento do fim;
  - Verificar se o deque está vazio;
  - Liberar a estrutura do deque;

Interface do tipo Deque

- **Vamos criar o arquivo deque.h, que representa a interface do deque:**

```
typedef struct deque Deque;
Deque *deque_cria (void); //igual ao código da fila
float deque_retira_fim (Deque *d);
float deque_retira_início (Deque *d); //igual ao código da fila
void deque_insere_fim (Deque *d, float v); //igual ao código da fila
void deque_insere_início (Deque *d, float v);
int deque_vazio (Deque *d); //igual ao código da fila
void deque_libera (Deque *d); //igual ao código da fila
```

unipampa
Universidade Federal do Pampa

Tópicos

- ~~O que é um Deque ?~~
- ~~Interface do tipo Deque .~~
- Implementação de Deque com vetor (contiguidade física).
- Implementação de Deque com listas encadeadas (alocação dinâmica).
- Resumo.

Implementação de Deque com Vetor

- Precisamos implementar duas novas funções:
  - **deque_insere_inicio** e **deque_retira_final;**
- Para inserir no início, precisamos inserir o elemento no índice que precede ini.
- Para retirar do final, devemos acessar a última posição do vetor preenchida. Podemos usar os campos, ini, n e TAM para isso

Implementação de Deque com Vetor

- A função para inserir um elemento no início do deque:

```
void deque_insere_inicio (Deque *d, float v){
        if(d->n == TAM){
                printf("Capacidade do deque estourou.\n");
                exit(1);
        }
        int ini = (d->ini – 1 + TAM) % TAM;
        d->vet[ini] = v;
        d->n++;
        d->ini = ini;
}
```

Implementação de Deque com Vetor

- A função para retirar um elemento do final do deque:

```
float deque_retira_final (Deque *d){
        float v;
        if(deque_vazio(d)){
                printf("Deque já vazio.\n");
                exit(1);
        }
        int ult = (d->ini+d->n-1)%TAM;
        v = d->vet[ult];
        d->n--;
        return v;

}
```

unipampa
Universidade Federal do Pampa

Tópicos

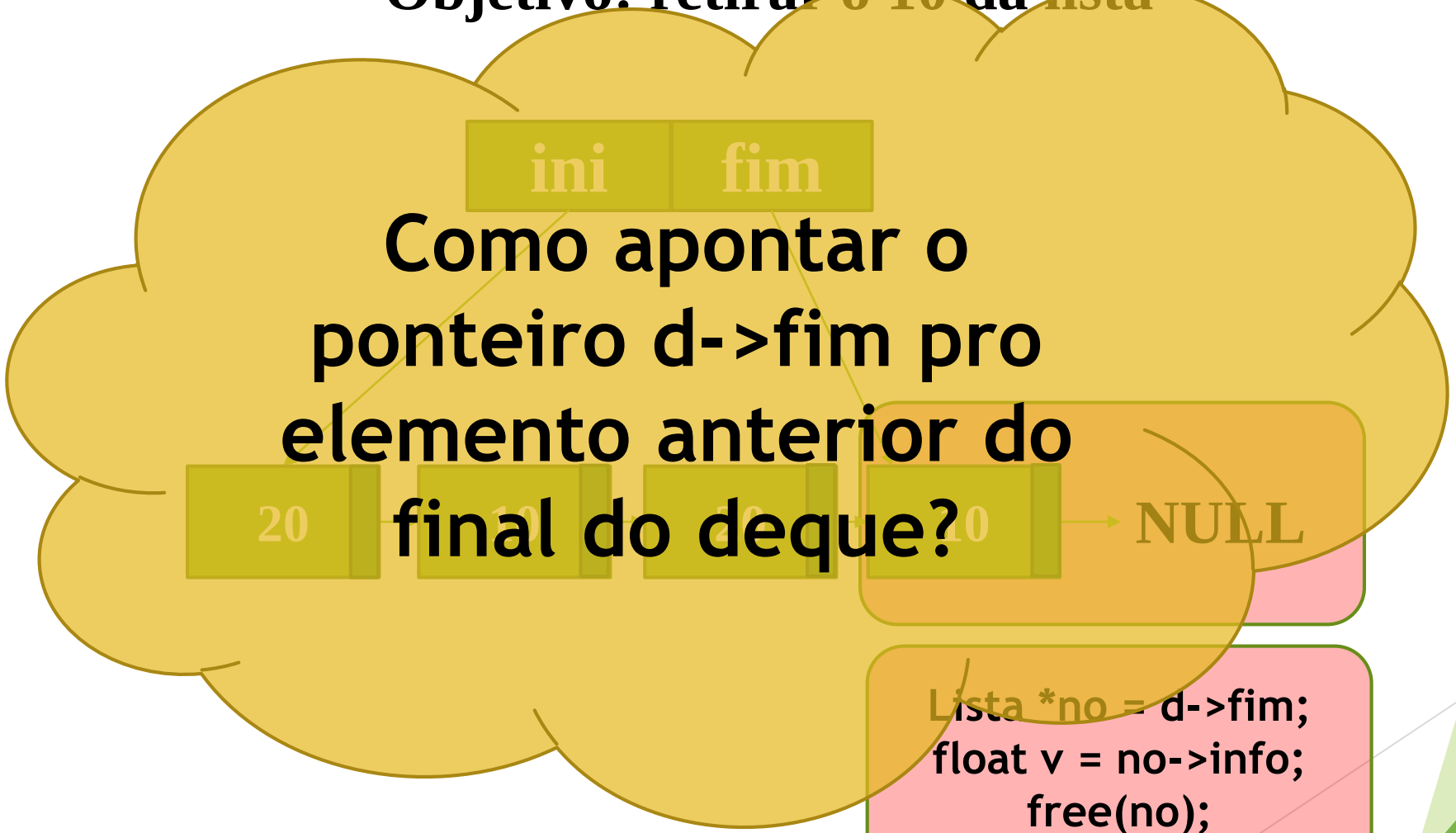
- ~~O que é um Deque ?~~
- ~~Interface do tipo Deque .~~
- ~~Implementação de Deque com vetor (contiguidade física).~~
- Implementação de Deque com listas encadeadas (alocação dinâmica).
- Resumo.

IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- Esta implementação merece uma discussão mais detalhada.
- Como retirar um elemento do final da lista com lista simplesmente encadeada?
- Todas outras funções seriam facilmente implementadas com listas encadeadas.

IMPLEMENTAÇÃO DE DEQUES COM LISTAS

Objetivo: retirar o 10 da lista



IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- Solução??

- **Lista Duplamente Encadeadas!!**

- **Com essa estrutura, temos acesso ao próximo elemento e ao elemento anterior do nó acessado;**

IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- Podemos ter a seguinte representação para o deque e a lista:

```
struct lista{
    float info;
    struct lista *ant;
    struct lista *prox;
};
typedef struct lista Lista;

struct deque{
    Lista *ini;
    Lista *fim;
};
```

IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- As funções para criar o deque, liberar e verificar se está vazio são praticamente iguais as funções de fila com lista, ficando como exercício implementa-las.

- O desafio é implementar as funções de inserção e remoção de elementos no início e final do deque.

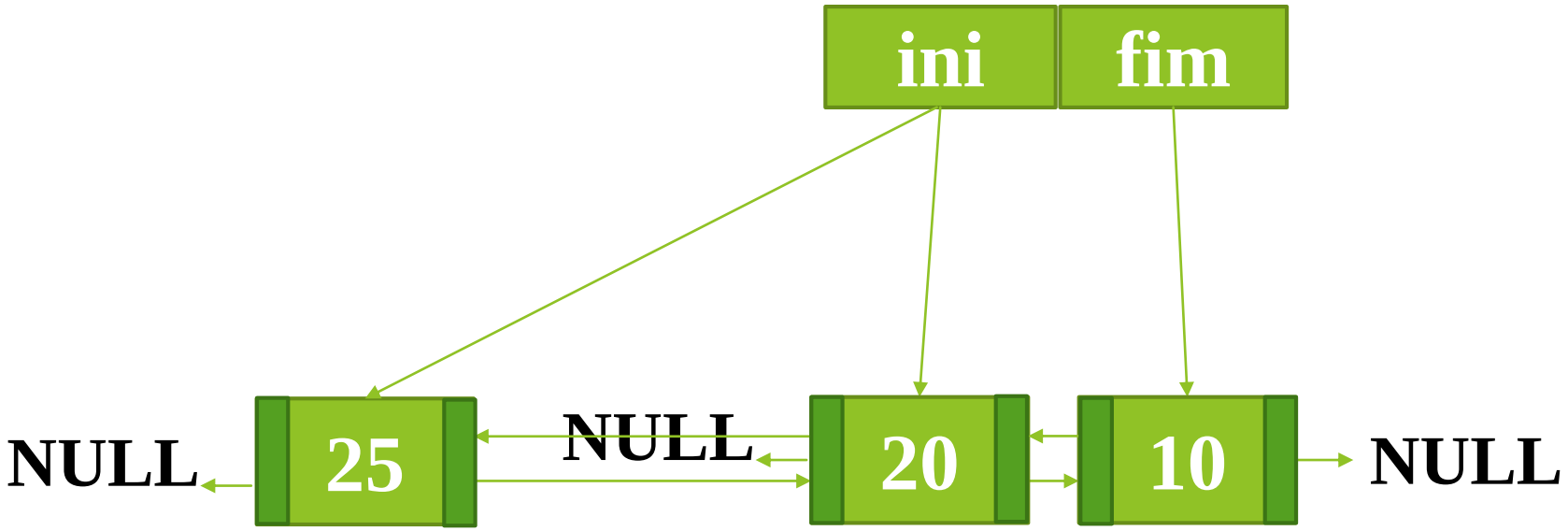
IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- A função para inserir um elemento no inicio do deque:

```
void deque_insere_inicio (Deque *d, float v){
        Lista *n= (Lista*) malloc(sizeof(Lista));
        n->info = v;
        n->prox = d->ini;
        n->ant = NULL;
        if(d->ini!=NULL){ //fila não estava vazia?
                d->ini->ant = n;
        }
        d->ini = n;
        if(d->fim==NULL){
                d->fim = d->ini;
        }

}
```

IMPLEMENTAÇÃO DE DEQUES COM LISTAS



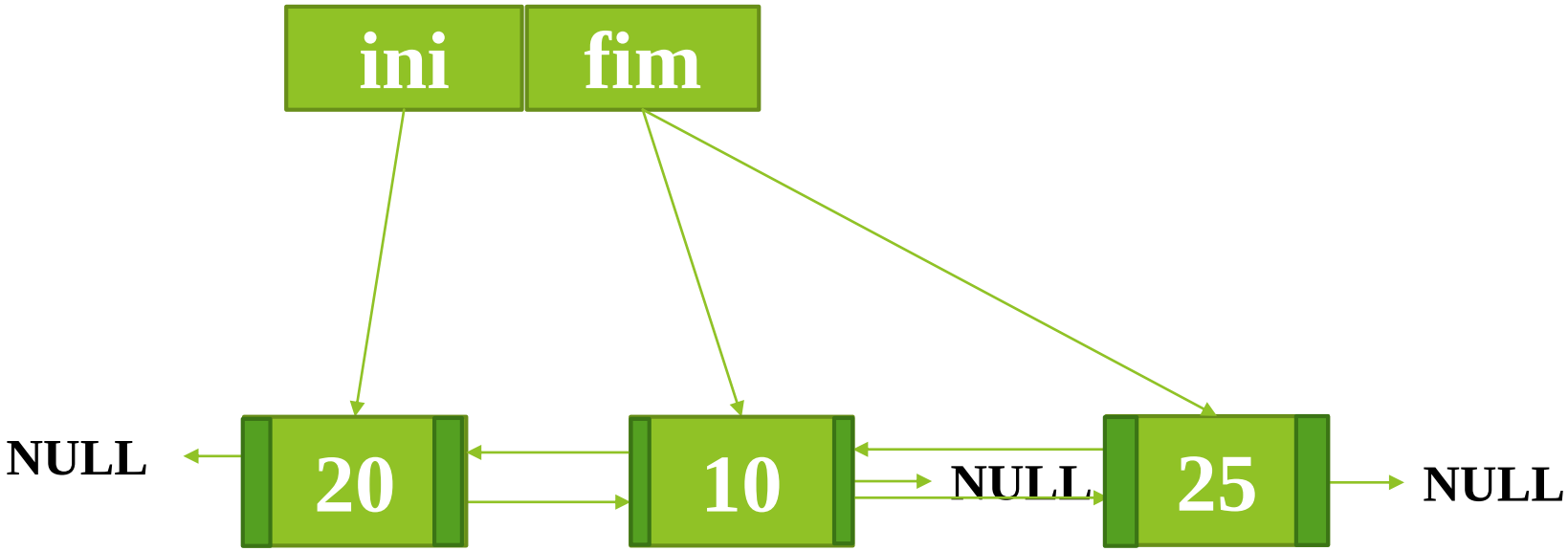
IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- A função para inserir um elemento no final do deque:

```
void deque_insere_final(Deque *d, float v){
        Lista *n= (Lista*) malloc(sizeof(Lista));
        n->info = v;
        n->prox = NULL;
        n->ant = d->fim;
        if(d->fim!=NULL){ //fila não estava vazia?
                d->fim->prox = n;
        }
        d->fim = n;
        if(d->ini==NULL){
                d->ini = d->fim;
        }

}
```

IMPLEMENTAÇÃO DE DEQUES COM LISTAS



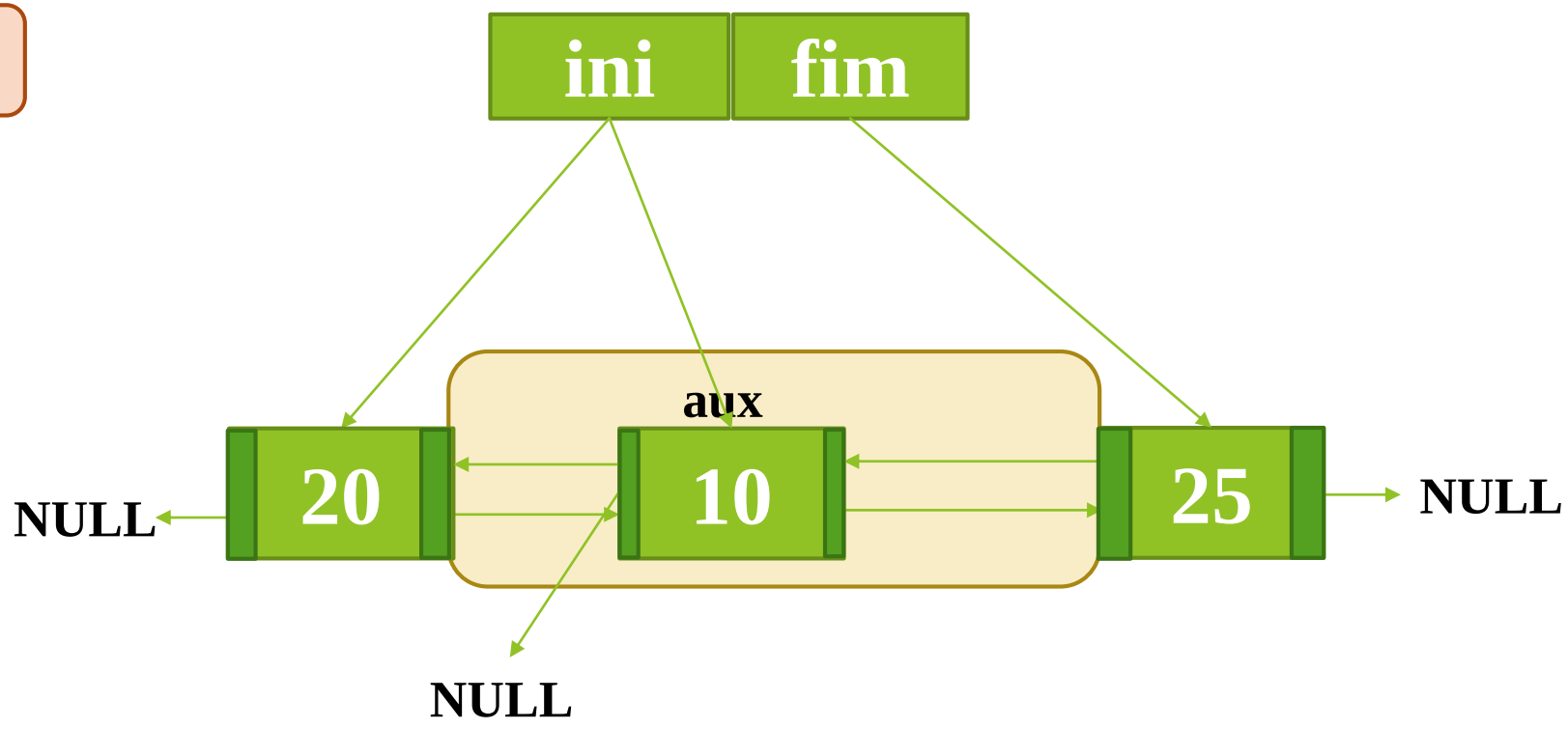
IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- A função para retirar um elemento do início do deque:

```
float deque_retira_inicio(Deque *d){
    if(deque_vazio(d)){ printf("Deque Vazio!.\n"); exit(1); }
    float v = d->ini->info;
    Lista *aux=d->ini->prox;
    if(aux != NULL){
            aux->ant=NULL;
    }
    free(d->ini);
    d->ini=aux;
    if(d->ini == NULL){
            d->fim = NULL;
    }
    return v;
}
```

IMPLEMENTAÇÃO DE DEQUES COM LISTAS

V = 20

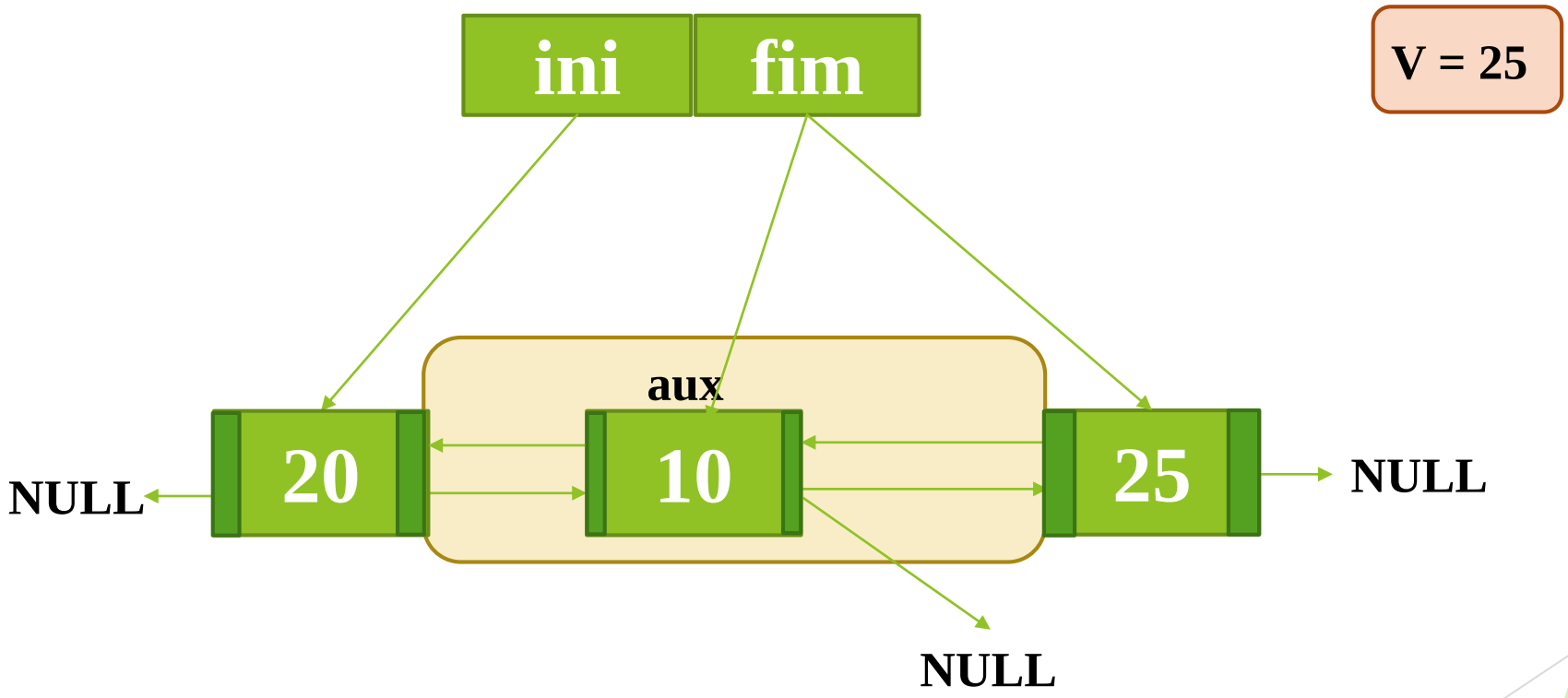


IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- A função para retirar um elemento do final do deque:

```
float deque_retira_final(Deque *d){
    if(deque_vazio(d)){ printf("Deque Vazio!.\n"); exit(1); }
    float v = d->fim->info;
    Lista *aux=d->fim->ant;
    if(aux != NULL){
            aux->prox=NULL;
    }
    free(d->fim);
    d->fim=aux;
    if(d->fim == NULL){
            d->ini = NULL;
    }
    return v;
}
```

IMPLEMENTAÇÃO DE DEQUES COM LISTAS



IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- A função para verificar se o deque está vazio:

```
int deque_vazio (Deque *d){
    return (d->ini == NULL); //1 = verdadeiro e 0 = falso
}
```

- A função para liberar a fila deve também liberar a lista:

```
void deque_libera(Deque *d){
    Lista *q = d->ini;
    while(q!=NULL){
        Lista *t = q->prox;
        free(q);
        q = t;
    }
    free(d);
}
```

IMPLEMENTAÇÃO DE DEQUES COM LISTAS

- Podemos ter uma função que mostre os elementos do deque, para fins de teste:

```
void deque_imprime(Deque *d){
    int i;
    for(i = 0, i < d->n; i++){
        printf("%f\n", d->vet[(f->ini+i)%TAMM]);
    }
}
void deque_imprime(Deque *d){
    Lista *q;
    for(q = d->ini; d!=NULL; q=q->prox){
        printf("%f\n", q->info);
    }
}
```

unipampa
Universidade Federal do Pampa

Tópicos

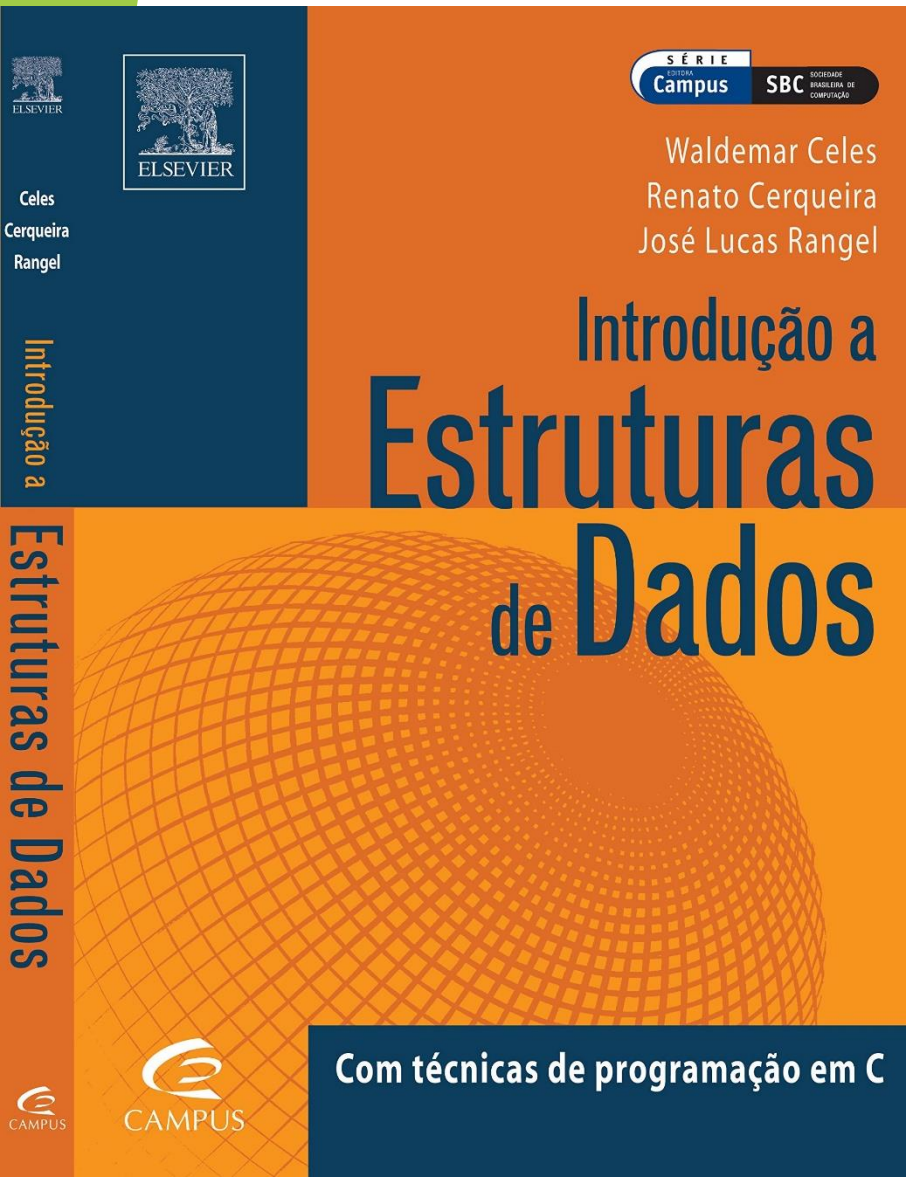
- ~~O que é um Deque ?~~
- ~~Interface do tipo Deque .~~
- ~~Implementação de Deque com vetor (contiguidade física).~~
- ~~Implementação de Deque com listas encadeadas (alocação dinâmica).~~
- Resumo.

Resumo

- **Foi demonstrado:**
  - **Funcionamento de um deque com vetor e lista;**
  - **Operações básicas e interface;**
  - **Exemplos em C;**

Dúvidas ?

Referências



- CELES, Waldemar; CERQUEIRA, Renato; RANGEL, José Lucas. **Introdução a Estruturas de Dados ~~e~~om técnicas de programação em C**. Rio de Janeiro: Elsevier (Campus), 2004. 4ª Reimpressão. 294 p.