

SIN 211 - Algoritmos e Estruturas de Dados (Deque)

Profº: Joelson Antônio dos Santos

Universidade Federal de Viçosa
Instituto de Ciências Exatas e Tecnológicas
Campus de Rio Paranaíba - MG

joelsonn.santos@gmail.com
Sala: BBT 233

10 de maio de 2018

Aula de Hoje

1 Deque

- Conceitos
- Implementação

Deque

- Deque (do inglês, *double-ended queue*).
- É uma lista linear que permite acesso aos elementos por duas extremidades.
- Pode simular filas ou pilhas.

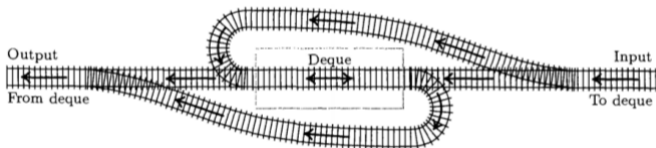


Figura: Estação de trem¹

¹The art of computer programming - Vol. 1 - D. Knuth

Deque

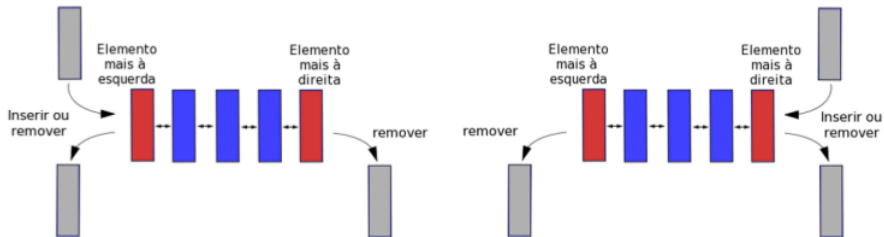
- Principais operações:
 - inicializar(D);
 - DequeVazio(D);
 - DequeCheio(D) // estático;
 - InserirInicio(D, x);
 - InserirFim(D, x);
 - RemoverInicio(D);
 - RemoverFim(D);
 - Imprimir(D);
 - LimparDeque(D);

Deque - Exemplo

Operações	Saída	Deque
inserirInicio(3)	-	(3)
inserirInicio(5)	-	(5,3)
removerInicio()	5	(3)
inserirFim(7)	-	(3,7)
removerInicio()	3	(7)
removerFim()	7	()
removerInicio()	"Deque Vazio"	()
inserirInicio(9)	-	(9)
inserirInicio(8)	-	(8,9)

Deque - Variações

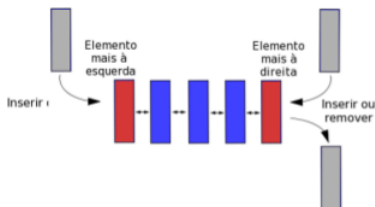
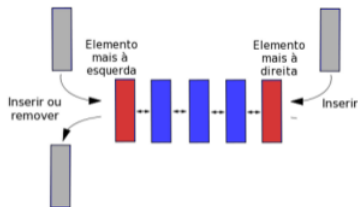
- *Input-restricted deque*
 - É possível remover elementos das duas extremidades, mas inserir em apenas uma delas.



Deque - Variações

- *Output-restricted deque*

- É possível inserir elementos das duas extremidades, mas remover em apenas uma delas.



Deque - Implementações

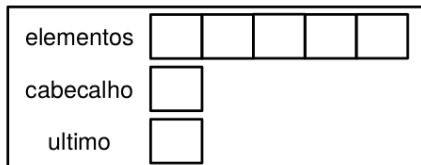
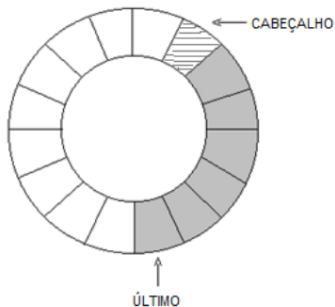
- Um deque pode ser implementado de maneira estática ou dinâmica.
 - Como poderíamos implementar de maneira estática?
 - Qual é a melhor representação para implementação de forma dinâmica?

Deque - Estático

- A implementação de deque estático segue o mesmo conceito de fila estática circular.
- Podemos usar as funções de remover do início e inserir no final da fila, porém devemos acrescentar as funções de inserir no início e remover no final.

Deque - Estático

- A estrutura permanece a mesma vista na fila estática.



Deque - Estático

- O “giro” pode acontecer:
 - No sentido anti-horário (inserir no final e remover no início).
 - No sentido horário (remover no final e inserir no início).

Deque - Dinâmico

- Pode ser implementado a partir do conceito de lista duplamente encadeada.
- Também pode ser criada uma estrutura para manipular os ponteiros de início e fim do deque.

Deque - Dinâmico

- Definição:

```
typedef struct sCell{  
    int info;  
    struct sCell *ant;  
    struct sCell *prox;  
}CELULA;  
  
typedef struct sDeque{  
    CELULA *inicio;  
    CELULA *fim;  
}DEQUE;
```

Deque - Operações

- Inserir no início:
 - Nome da função: `inserirInicio`;
 - Tipo de retorno: **int**;
 - Número de parâmetros: 2;
 - 1 Variável do tipo `DEQUE`;
 - 2 Variável do tipo: `{tipo_informação}`;
 - Descrição: função responsável por criar, inicializar e inserir um elemento no início do deque.
 - Levar em consideração os seguintes casos:
 - Não foi possível alocar memória para o novo elemento;
 - O deque está vazio;
 - O deque não está vazio;

Deque - Operações

- Remover fim:
 - Nome da função: `removerFim`;
 - Tipo de retorno: `{tipo_informação}`;
 - Número de parâmetros: 1;
 - 1 Variável do tipo `DEQUE`;
 - Descrição: função responsável por remover um elemento no final do deque.
 - Considerar as seguintes situações
 - O deque está vazio (retornar mensagem de erro);
 - Depois de remover o elemento o deque ficou vazio;
 - Depois de remover o elemento o deque não ficou vazio;

Deque - Operações

- Quais são as outras operações?
- Como ficariam as implementações?
- É possível acessar elementos no meio de um deque?
- Como é feita a impressão de todos os elementos de um deque?
- Como pode ser feita a modificação de um elemento na posição k de um deque?

Deque - Complexidade

- Quais os custos computacionais Big-O para as seguintes operações de um deque:
- Inserir e remover no início? E no final?
- Acessar o i -ésimo elemento em um deque.
- Verificar a quantidade de elementos em um deque. É possível otimizar isso? Como?
- Impressão de um deque.
- Inversão de valores de um deque. Ex: 2,3,4,5 para 5,4,3,2.

Bibliografia Básica

- Deque estático: <https://goo.gl/YNzYHX>
Acessado em: 16/10/2017
- *Donald Knuth. The Art of Computer Programming, Volume 1: Fundamental Algorithms, Third Edition. Addison-Wesley, 1997. ISBN 0-201-89683-4. Section 2.2.1: Stacks, Queues, and Deques, pp. 238–243.*