

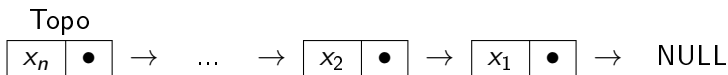
Pilhas e Filas Encadeadas

Prof. Jefferson T. Oliva

Algoritmos e Estrutura de Dados 2 (AE43CP)
Engenharia de Computação
Departamento Acadêmico de Informática (Dainf)
Universidade Tecnológica Federal do Paraná (UTFPR)
Campus Pato Branco

- Pilhas Encadeadas
- TAD Pilha Encadeadas
- Filas Encadeadas
- TAD Filas Encadeadas

- Pilhas encadeadas
 - Alocação dinâmica
 - Estrutura similar ao da lista encadeada
 - A principal mudança está nas operações de inserção e de remoção



Introdução

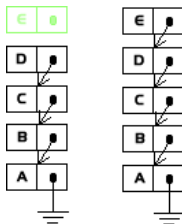
- Filas encadeadas
 - Alocação dinâmica
 - Estrutura similar ao da lista/pilha encadeada
 - A principal mudança está nas operações de inserção e de remoção
 - Inserção no final
 - Remoção no início



Pilhas Encadeadas

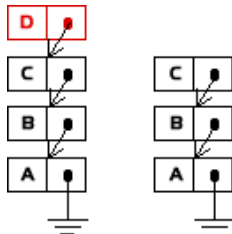
Pilhas Encadeadas

- Não há necessidade do uso de cursores para indicar o topo da pilha
- Para Empilhar um novo item, basta criar uma nova célula e ligar o seu cursor no topo da pilha



Pilhas Encadeadas

- Para desempilhar, basta desligar a célula que está no topo



TAD Pilhas Encadeadas

- Operações básicas
 - Criar uma pilha
 - Verificar se a pilha está vazia
 - Empilhar
 - Desempilhar
 - Imprimir
 - Liberar

- Exemplo de arquivo .h para estrutura do tipo pilhaE:

```
typedef struct PilhaE PilhaE;  
  
PilhaE* criar_pilhaE();  
  
int pilhaE_vazia(PilhaE *p);  
  
void empilhar(int key, PilhaE *p);  
  
int desempilhar(PilhaE *p);  
  
int acessar_valor_topo(PilhaE *p);  
  
void imprimir_pilha(PilhaE *p);  
  
int liberar_pilha(PilhaE *p);
```

- A implementação do código-fonte do TAD da pilha encadeada está disponível no repositório da disciplina no GitHub

- Complexidade das operações empilhar e desempilhar:

Operação	Complexidade
Empilhar	$\Theta(1)$
Desempilhar	$\Theta(1)$

Nas operações acima, a complexidade para o melhor caso é de $\Omega(1)$ e para o pior caso, de $O(1)$

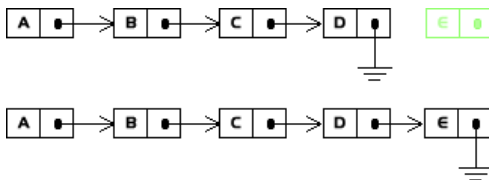
Logo, a complexidade das operações acima é de $\Theta(1)$

Filas Encadeadas

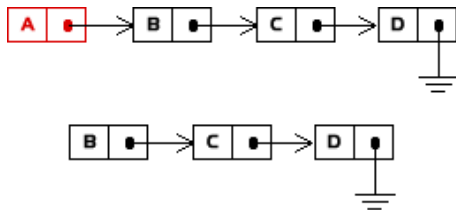
Filas Encadeadas

Filas Encadeadas

- O item na primeira posição da lista é o primeiro da fila
- A inclusão de um novo elemento é posicionado no final da fila, após a estrutura ser percorrida (em uma simples implementação)



- Para desenfilear um elemento, basta desligar a primeira célula



TAD Filas Encadeadas

- Operações básicas
 - Criar uma fila
 - Verificar se a fila está vazia
 - Enfileirar
 - Desenfileirar
 - Imprimir
 - Liberar

- Exemplo de arquivo .h para estrutura do tipo filaE:

```
typedef struct FilaE FilaE;  
  
FilaE* criar_filaE();  
  
int filaE_vazia(FilaE* f);  
  
void enfileirar(int key, FilaE* f);  
  
int desenfileirar(FilaE* f);  
  
void imprimir(FilaE* f);  
  
int verificar_inicio(FilaE *f);  
  
int verificar_fim(FilaE *f);  
  
int liberar_filaE(FilaE* f);
```

- A implementação do código-fonte do TAD da fila encadeada está disponível no repositório da disciplina no GitHub

- Complexidade das operações enfileirar e desenfileirar:

Operação	Complexidade
Enfileirar	$\Theta(1)$
Desenfileirar	$\Theta(1)$

Nas operações acima, a complexidade para o melhor caso é de $\Omega(1)$ e para o pior caso, de $O(1)$

Logo, a complexidade das operações acima é de $\Theta(1)$



Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C.
Introduction to Algorithms.
Third edition, The MIT Press, 2009.



Cozman, F. G.
Pilhas, Filas, Listas. PMR2300 – Estrutura de Dados.
Slides. Engenharia Mecatrônica. Poli/USP/São Paulo, 2019.



Oliva, J. T.
Pilhas Encadeadas. AE22CP – Algoritmos e Estrutura de
Dados I.
Notas de Aula. Engenharia de Computação.
Dainf/UTFPR/Pato Branco, 2019.



Oliva, J. T.

Filas Encadeadas. AE22CP – Algoritmos e Estrutura de Dados I.

Notas de Aula. Engenharia de Computação.
Dainf/UTFPR/Pato Branco, 2019.



Szwarcfiter, J.; Markenzon, L.

Estruturas de Dados e Seus Algoritmos.
LTC, 2010.



Tenenbaum, A.; Langsam, Y.

Estruturas de Dados usando C.
Pearson, 1995.



Ziviani, N.

Projeto de Algoritmos - com implementações em Java e C++.
Thomson, 2007.