

CENTRO UNIVERSITÁRIO DA FEI

BRENNO TONDATO DE FARIA

ATIVIDADE 1 - PEL216

São Bernardo do Campo

2019

BRENNO TON DATO DE FARIA

ATIVIDADE 1 - PEL216

Realatório da Atividade proposta da disciplina Programação Científica (PEL216) ministrada pelo Prof. Dr. Reinaldo Bianchi

São Bernardo do Campo

2019

RESUMO

Nesta atividade é apresentada os conceitos de estrutura de dados computacionais Pilha e Fila.

Keywords: Estrutura de Dados, Pilha, Fila

ABSTRACT

This activity aims to present the concepts of data structures as Stack and Queue.

Keywords: Data Structure, Stack, Queue

LISTA DE ILUSTRAÇÕES

Ilustração 1 – Analise de Complexidade de Diversos Tipos de Estruturas de Dados	6
Ilustração 2 – Método LIFO	7
Ilustração 3 – Método LIFO	8
Ilustração 4 – Diagrama de Classe	9
Ilustração 5 – Sequencia de testes Pilha	10
Ilustração 6 – Sequencia de testes Fila	11

SUMÁRIO

1	Introdução	6
2	Conceitos Fundamentais	7
3	Metodologia	9
4	Experimentos	10
5	Conclusão	12
	REFERÊNCIAS	13

1 INTRODUÇÃO

Estruturas de dados são largamente utilizadas em sistemas computacionais. Dentre muitas estruturas duas das mais antigas Pilhas e Filas. independente de suas idades pilhas e filas são largamente utilizadas até hoje devido á sua complexidade no acesso e na remoção de elementos. A Figura 1 a seguir ilustra a complexidade destas estrutura, comprada à outras.

Figura 1 – Analise de Complexidade de Diversos Tipos de Estruturas de Dados

Common Data Structure Operations									
Data Structure	Time Complexity								Space Complexity
	Average				Worst				Worst
	Access	Search	Insertion	Deletion	Access	Search	Insertion	Deletion	
Array	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$
Stack	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$
Queue	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$
Singly-Linked List	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$
Doubly-Linked List	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$	$\theta(n)$	$\theta(1)$	$\theta(1)$	$\theta(n)$
Skip List	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n \log(n))$
Hash Table	N/A	$\theta(1)$	$\theta(1)$	$\theta(1)$	N/A	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$
Binary Search Tree	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$
Cartesian Tree	N/A	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	N/A	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$
B-Tree	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$
Red-Black Tree	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$
Splay Tree	N/A	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	N/A	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$
AVL Tree	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$
KD Tree	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(\log(n))$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$	$\theta(n)$

(Do Autor, 2019)

A estrutura de dados pilha teve sua origem em meados de 1940. inicialmente seu conceito foi exposto nos trabalhos de (KONRAD...,) no ano de 1944 e em 1945 já estava sendo aplicada em trabalhos de Alan M. Turing (CARPENTER; DORAN, 1977) porém com seus conteitos estavam sendo expressos pelo nome de *bury* e *unbury*.

Já a estrutura de Fila não se sabe ao certo quando teve sua origem , porém esta estrutura é uma das mais utilizadas em redes de comunicação. Uma das teorias em redes e comunicação que mais faz uso desta estrutura é a teoria das filas (ERLANG, 1909).

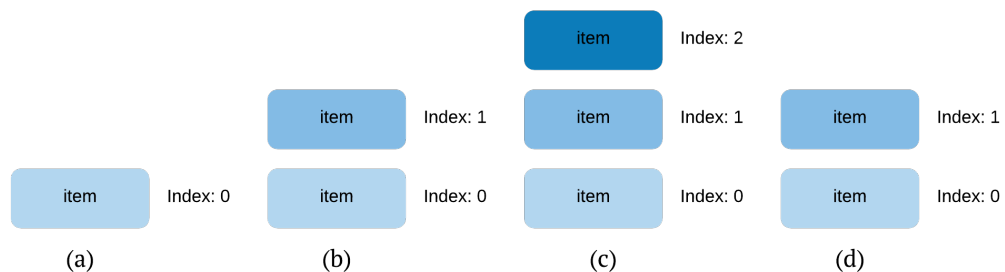
Neste trabalho as estruturas de dados pilha e fila são utilizadas afim de compreender seus fundamentos.

2 CONCEITOS FUNDAMENTAIS

Pilhas e Filas estão entre as estruturas de dados mais utilizadas. Estas estruturas possuem seus próprios mecanismos de acesso sendo estes o método *Last In First Out (LIFO)* para Pilhas e *First in First Out (FIFO)* Para Filas.

O método LIFO de acesso a Pilhas consiste na retirada do ultimo elemento inserido na estrutura. Um exemplo pode ser observado na Figura 2 a seguir.

Figura 2 – Método LIFO



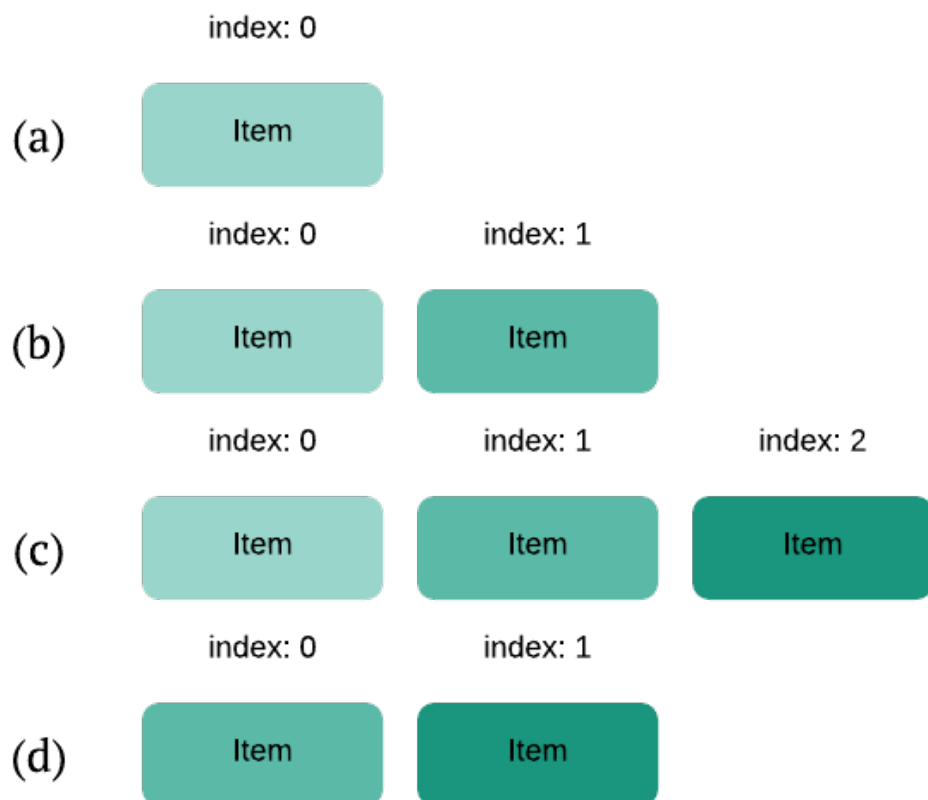
(Do Autor, 2019)

Através desta figura é possível observar o crescimento o índice (*index*) conforme a quantidade de itens são inseridos. Quando é realizada uma exclusão de item o de maior índice é retirado, como pode ser observado na Figura 2 (d).

O Método FIFO de acesso a Filas ocorre de maneira contrária ao método das Pilhas. Neste acesso o primeiro elemento adicionado à estrutura será o primeiro a ser retirado. O comportamento da estrutura pode ser observado na Figura 3 a seguir.

Pode-se observar o crescimento do índice conforme novos item são adicionados, porém desta vez quando há uma operação de exclusão o primeiro item é retirado como mostra a Figura 3 (d).

Figura 3 – Método LIFO



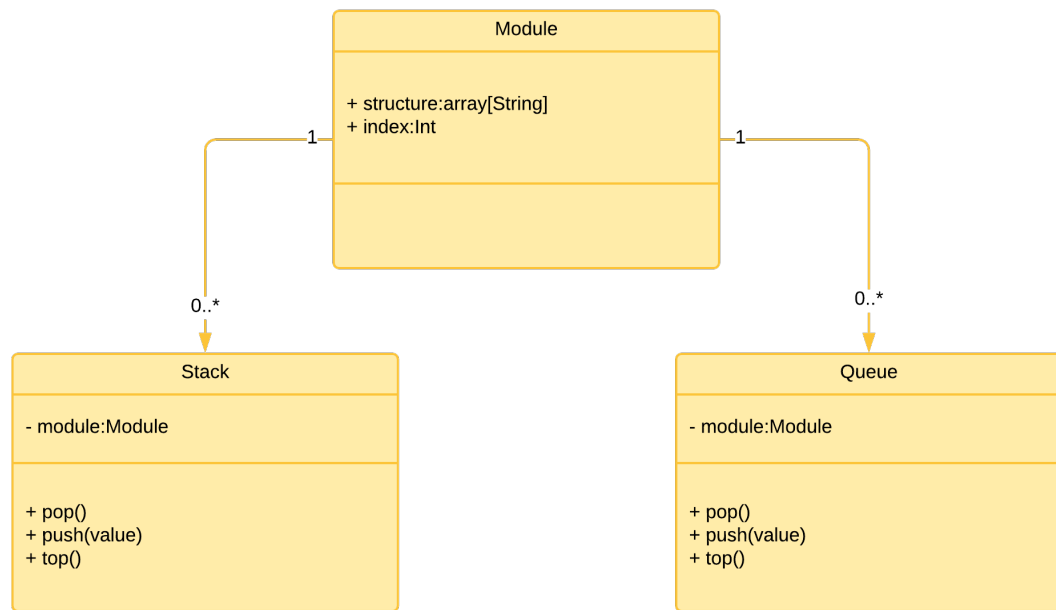
(Do Autor, 2019)

3 METODOLOGIA

Para a realização deste trabalho foi elaborado um sistema orientado a objetos para testar os conceitos de Pilhas e Filas.

Este sistema possui o diagrama de classe mostrado na Figura 4.

Figura 4 – Diagrama de Classe



(Do Autor, 2019)

Neste diagrama é possível observar uma classe sem método algum. Esta classe chamada *Module* se refere á estrutura basica de uma Pilha e uma Fila sendo esta estrutura um indice para controlar o sequenciamento dos item e um *array* para armazenar os item. Sendo assim cada tipo de estrutura de dados possui seus próprios métodos para controle dos itens.

4 EXPERIMENTOS

A Figura 5 representa a sequência de testes realizados com a classe Pilha.

Figura 5 – Sequencia de testes Pilha

```

===== STACK TESTS =====
Stack is ["brenno"]
Stack is ["brenno", "ellen"]
Stack is ["brenno", "ellen", "tadeu"]
Stack is ["brenno", "ellen", "tadeu", "mariana"]
Stack is ["brenno", "ellen", "tadeu", "mariana", "bianca"]
Stack is ["brenno", "ellen", "tadeu", "mariana", "bianca", "paula"]
Stack Top Value : paula
Stack is ["brenno", "ellen", "tadeu", "mariana", "bianca"]
Stack Top Value : bianca
Stack is ["brenno", "ellen", "tadeu", "mariana"]
Stack is ["brenno", "ellen", "tadeu"]
Stack is ["brenno", "ellen"]
Stack is ["brenno"]
Stack is []
Stack is empty []
=====

```

(Do Autor, 2019)

Nestes testes é possível observar a quantidade de elementos crescendo durante sucessivas inserções. Quando é solicitado o topo *Top* da pilha é devolvido o último elemento da estrutura e quando é realizada uma operação de remoção é o último elemento que é retirado.

A Figura 6 representa a sequência de testes realizados com a classe Fila.

Nestes testes é possível observar a quantidade de elementos crescendo durante sucessivas inserções. Quando é solicitado o topo *Top* da estrutura, o primeiro elemento é retornado e

Figura 6 – Sequencia de testes Fila

```
===== QUEUE TESTS =====  
  
Queue is ["Brenno"]  
Queue is ["Brenno", "Ellen"]  
Queue is ["Brenno", "Ellen", "Tadeu"]  
Queue Top Value : Brenno  
Queue is ["Ellen", "Tadeu"]  
Queue is ["Tadeu"]  
Queue Top Value : Tadeu  
Queue is []  
Queue is empty []  
  
=====
```

(Do Autor, 2019)

quando uma operação de exclusão é realizada o primeiro elemento é retirado e os índices são redistribuídos.

5 CONCLUSÃO

O presente trabalho se comprometeu a realizar a implementação de estrutura de dados Pilha e Fila afim de compreender seus fundamentos. Através dos testes pode-se comparar o comportamento das duas estruturas e assim compreender seus fundamentos. Uma possível extensão deste trabalho seria a implementação destas duas estruturas em outros tipos de computação como por exemplo computadores biológicos ou até mesmo quânticos.

REFERÊNCIAS

CARPENTER, B. E.; DORAN, R. W. The other Turing machine. **The Computer Journal**, v. 20, n. 3, p. 269–279, 01 1977. ISSN 0010-4620. Disponível em: <<https://doi.org/10.1093/comjnl/20.3.269>>.

ERLANG, A. K. The theory of probabilities and telephone conversations. **Nyt. Tidsskr. Mat. Ser. B**, v. 20, p. 33–39, 1909.

KONRAD Zuse. Disponível em: <https://en.wikipedia.org/wiki/Konrad_Zuse>.