

Pilhas e Filas

Filas

Prof. Edson Alves – UnB/FGA

1. Filas
2. Filas na STL

Filas

Definição de fila

- Uma fila é um tipo de dados abstrato cuja interface define que o primeiro elemento inserido na pilha é o primeiro a ser removido
- Esta estratégia de inserção e remoção é denominada FIFO – *First In, First Out*
- De acordo com sua interface, uma fila não permite acesso aleatório ao seus elementos (apenas o elemento do topo da fila pode ser acessado)
- As operações de inserção e remoção devem ter complexidade $O(1)$

Interface de uma fila

Método	Complexidade	Descrição
clear(F)	$O(N)$	Esvazia a fila F, removendo todos os seus elementos
empty(F)	$O(1)$	Verifica se a fila F está vazia ou não
push(F, x)	$O(1)$	Insere o elemento x no final da fila F
pop(F)	$O(1)$	Remove o elemento que está no início da fila F
front(F)	$O(1)$	Retorna o elemento que está no início da fila F
size(F)	$O(1)$	Retorna o número de elementos armazenados na fila F

Exemplo dos métodos da interface de uma fila

Método

Retorno

`empty(F)`

Fila

Exemplo dos métodos da interface de uma fila

Método

Retorno

`empty(F)`

`True`

Fila

Exemplo dos métodos da interface de uma fila

Método

Retorno

push(F, 5)

Fila

Exemplo dos métodos da interface de uma fila

Método

Retorno

push(F, 5)

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

`push(F, 11)`

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

`push(F, 11)`

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

push(F, 7)

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

push(F, 7)

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

pop(F)

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

pop(F)

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

`size(F)`

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

size(F)

2

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

front(F)

Fila



Exemplo dos métodos da interface de uma fila

Método

Retorno

front(F)

11

Fila



Implementação de uma fila

- Como uma fila é um tipo de dados abstrato, ela não impõe nenhuma restrição quanto à sua implementação
- É possível implementar uma fila por composição, usando listas encadeadas ou uma deque (*double-ended queue*)
- A estratégia FIFO precisa de operações de inserção e remoção eficientes nos dois extremos do contêiner, o que inviabiliza o uso do vector e da `forward_list`
- Se há uma estimativa do tamanho máximo de elementos a serem inseridos na fila, é possível usar um *array* estático e o mesmo princípio de uma lista circular para implementar uma fila

Exemplo de implementação de fila em C++

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 template<typename T, size_t N>
6 class Queue {
7 public:
8     Queue() : first(0), last(0), _size(0) {}
9
10    void clear() { first = last = _size = 0; }
11    bool empty() const { return _size == 0; }
12    size_t size() const { return _size; }
13
14    void push(const T& x) {
15        if (_size == N) throw "Fila cheia";
16
17        elems[last] = x;
18        last = (last + 1) % N;
19        _size++;
20    }
```

Exemplo de implementação de fila em C++

```
22 void pop()
23 {
24     if (_size == 0) throw "Lista vazia";
25
26     first = (first + 1) % N;
27     _size--;
28 }
29
30 const T& front() const
31 {
32     if (_size == 0) throw "Lista vazia";
33
34     return elems[first];
35 }
36
37 private:
38     array<T, N> elems;
39     int first, last;
40     size_t _size;
41 };
```

Exemplo de implementação de fila em C++

```
43 int main() {  
44     Queue<int, 10> q;  
45  
46     cout << "Empty? " << q.empty() << '\n';  
47  
48     for (int i = 1; i <= 10; ++i) q.push(i);  
49  
50     cout << "Size = " << q.size() << '\n';  
51     cout << "Front = " << q.front() << '\n';  
52  
53     for (int i = 0; i < 5; ++i) q.pop();  
54  
55     q.push(11);  
56     q.push(12);  
57  
58     cout << "Size = " << q.size() << '\n';  
59     cout << "Front = " << q.front() << '\n';  
60  
61     return 0;  
62 }
```

Filas na STL

- A STL do C++ oferece uma implementação de fila: a classe `queue`
- Assim como no caso das pilhas, o contêiner usado na composição é, por padrão, a `deque`
- Este contêiner pode ser substituído por qualquer contêiner que contenha os métodos `pop_front()`, `push_back()` e `size()`, dentre outros
- O método `swap()` também está disponível

Exemplo de uso de fila em C++

```
1 #include <bits/stdc++.h>
2
3 int main()
4 {
5     std::queue<std::string> q { "um", "dois", "tres" };
6
7     std::cout << "Vazia? " << q.empty() << '\n';           // Falso
8
9     std::cout << "Primeiro elemento: " << q.front() << '\n'; // "um"
10    std::cout << "Último elemento: " << q.back() << '\n';    // "tres"
11
12    q.push("quatro");
13
14    std::cout << "Tamanho da fila: " << q.size() << '\n';    // 4
15
16    q.pop();
17    std::cout << "Primeiro elemento: " << q.front() << '\n'; // "dois"
18
19    return 0;
20 }
```

1. **DROZDEK**, Adam. *Algoritmos e Estruturas de Dados em C++*, 2002.
2. **KERNIGHAN**, Bryan; **RITCHIE**, Dennis. *The C Programming Language*, 1978.
3. **STROUSTROUP**, Bjarne. *The C++ Programming Language*, 2013.
4. C++ Reference. [Queue](#), acesso em 22/04/2019.