

1) Utilizando uma estrutura do tipo vetor, implemente em linguagem C um exemplo de estrutura do tipo fila circular, desenvolvam as funções de inserir, remover, listar e procurar um elemento da fila.

Uma estrutura fila é igual uma fila na vida real, o primeiro que chega sempre é o primeiro que sai é sempre o último que chega é o último que sair porque vai ficar no final da fila.

Por exemplo na fila do banco a primeira pessoa a entrar no final da fila será a primeira pessoa a chegar na frente da fila.

Exemplos de filas:

Pacote de dados esperando para serem transmitidos pela rede, por exemplo 1 mb precisa dividir em 4 pacotes de 250kb, então eles são enfileirados e enviados em fila e quando chega no outro computador é preciso respeitar a ordem da fila pra não quebrar a sequência dos dados.

Outro exemplo clássico que vimos na aula de Raul é a fila de impressora, no qual os serviços de impressão aguarda a impressora ficar disponível. A primeira pessoa que clicar em imprimir terá o documento impresso primeiro e a última pessoa terá a impressão feita por último.

Com relação às operações

- 1 - Inserir (colocar um item sempre no final da fila respeitando a ordem)
- 2 - remover (remova sempre o item do início da fila respeitando a ordem)
- 3 - listar (exibe os elementos inseridos na fila)
- 4 - buscar (elemento) (procura e retorna se existe ou não o elemento informado na fila)

```
import numpy as np
class FilaCircular:
    # a fila recebe a como parâmetro capacidade de armazenamento fixa
    def __init__(self, cap):
        self.cap = cap
        self.inicio = 0 # atributo de controle (início da fila)
        self.final = -1 # atributo de controle (fim da fila)
        self.num_el = 0 # numero de elementos da fila começa com 0
        #python usa lista como padrão, os valores são do tipo int array da
lib numpy
        self.num = np.empty(self.cap, dtype=int)
        #metodo para verificar se a flia está vazia
    def __fila_vazia(self):
        return self.num_el == 0
        #metodo que verifica se a fila esta cheia
    def __fila_cheia(self):
        return self.num_el == self.cap
        #metodo p inserir o elemento da fila (recebe num como parâmetro)
    def inserir(self, num):
        if self.__fila_cheia(): # antes de colocar verifica se a fila ta
cheia
```

```

        print('nao cabe mais elementos na fila')
        return # sair da funcao

#verificar se o final da fila está para esquerda ou para direita
if self.final == self.cap - 1:
    self.final = -1 # se o final da fila chegou ao final do vetor o
final decrementa -1
    self.final += 1 # se não for o final da fila então incrementa + 1
no final do vetor
    self.num[self.final] = num # o final da fila recebe o numero
    self.num_el += 1 # incrementar o numero de elementos + 1

def remover(self):
    if self.__fila_vazia(): # verificar se a fila ta vazia
        print('nao tem nada na fila')
        return # sai do metodo
    # sempre tira o elemento que esta no inicio da fila
    temp = self.num[self.inicio] # variavel temporaraia que indica qual
elemento foi desenfileirado
    self.inicio += 1 # estou trocando a posicao inicial da fila
    if self.inicio == self.cap - 1: # se o inicio da fila esta no final
do vetor
        self.inicio = 0 # o inicio rerebe a posicao (0 inicial)
    self.num_el -= 1 # decrementa o numero de elementos da fila
    return temp # retorna o elemento retirado da fila

def listar(self):
    if self.__fila_vazia(): #verificar se a fila nao tem naada
        return -1 # sair do metodo
    for i in range(self.num_el): # laço p percorrer a fila
        print(self.num[i]) # exibir elemento da fila em ordem

def buscar(self, num):
    self.Num = num
    if self.__fila_vazia(): #verificar se a fila nao tem naada
        return -1 # sair do metodo
    for i in range(self.num_el): # laço p percorrer a fila
        if self.num[i] == self.Num:
            print("Este numero existe na fila: ", self.Num )
        else:
            print("este numero nao existe na fila")

if __name__ == "__main__": # funcao principal + testes

```

```

    fila = FilaCircular(5)
    fila.inserir(3)
    fila.inserir(4)
    fila.inserir(5)
    fila.listar()
    fila.remover()
    fila.remover()
    print("\n")
    fila.listar()
    fila.buscar(3)
    fila.buscar(56)
    fila.inserir(5)
    fila.inserir(8)
    fila.inserir(16)
    print("\n")
    fila.listar()
>> Saída:
3
4
5

3
Este número existe na fila: 3
este numero nao existe na fila

16
4
5
5

```

Referências:

Slides das aulas teóricas do Prof. Raul Benites Paradedda (Estrutura de Dados 2020)

O código-fonte está disponível e rodando no repositório do github disponível em:

<https://github.com/abnermuxah/esturua-de-dados>

Aulas teóricas do prof Jones (IA Expert)

<https://iaexpert.academy/courses/estrutura-de-dados-e-algoritmos-python-guia-completo/>