Estrutura de Dados

Prof^a. Dr^a. Alana Morais alanamm.prof@gmail.com

Objetivo da aula

Entender o comportamento das filas

Recapitulando - Estrutura de dados

Dados simples:

- padrão:
 - inteiro (int);
 - real (float);
 - caracter (str);
 - lógico (boolean).

Dados estruturados:

- Estáticos:
 - arrays;
 - registros;
 - arquivos;
 - conjuntos;
 - o cadeias.
- Dinâmicos:
 - filas;
 - o pilhas;
 - listas encadeadas;
 - árvores;
 - o grafos.

Recapitulando - Estrutura de dados

Dados simples:

- padrão:
 - o inteiro (int);
 - real (float);
 - caracter (str);
 - lógico (boolean).

Dados estruturados:

- Estáticos:
 - arrays;
 - registros;
 - arquivos;
 - conjuntos;
 - o cadeias.
- Dinâmicos:
 - filas;
 - o pilhas;
 - listas encadeadas;

listas ou vetores

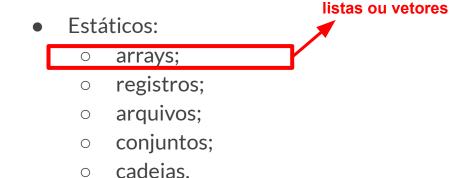
- árvores;
- o grafos.

Recapitulando - Estrutura de dados

Dados simples:

- padrão:
 - inteiro (int);
 - real (float);
 - caracter (str);
 - lógico (boolean).

Dados estruturados:



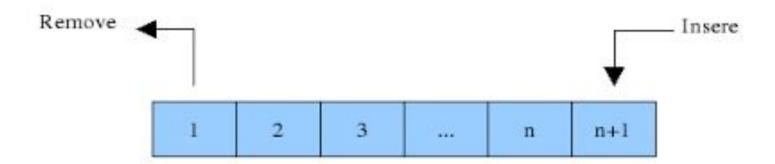
Dinâmicos:

o filas;
o pilhas;
o listas encadeadas;
o árvores;
o grafos.

O que são filas?



O que são filas?



O que são filas?



Operações

- Criar fila
- Mostrar valores da fila
- Inserir valores
- Remover valores
- Definir tamanho da fila

Operações

- Criar fila
- Mostrar valores da fila
- Inserir valores
- Remover valores
- Definir tamanho da fila

Como criar uma fila em Python?



```
class Fila(object):
   def __init__(self):
     self.dados = [ ]
```

```
class Fila(object):
   def __init__(self):
      self.dados = [ ]
```

Classe construtora

Por que preciso usar uma classe para definir uma Fila?



Por que preciso usar uma classe para definir uma Fila?

Para acessar diversas vezes em vários códigos Python

fila.py

```
class Fila(object):
    def __init__(self):
        self.dados = [ ]

fila1 = Fila()
```

```
fila.py
                                             programa.py
class Fila(object):
                                   import fila
    def __init__(self):
         self.dados = [ ]
                                   fila1 = fila.Fila()
fila1 = Fila()
```

Exercício A

Crie uma classe chamada Fila e inicialize essa classe: a) dentro do arquivo Python da classe e b) em outro arquivo Python

Operações

- Criar fila
- Mostrar valores da fila
- Inserir valores
- Remover valores
- Definir tamanho da fila

Mostrar valores da fila

fila.py

```
class Fila(object):
    def __init__(self):
        self.dados = [ ]

    def getFila(self):
        return self.dados
```

Mostrar valores da fila

```
fila.py
class Fila(object):
    def init (self):
        self.dados = [ ]
    def getFila(self):
        return self.dados
```

programa.py

```
import fila

filaTeste = fila.Fila()

print(filaTeste.getFila())
```

Exercício B

Crie uma classe chamada Fila com os seguintes valores: -1, -2, -3, -4. Em seguida, mostre esses valores na tela.

Operações

- Criar fila
- Mostrar valores da fila
- Inserir valores
- Remover valores
- Definir tamanho da fila

Inserir valores na fila

fila.py

```
class Fila(object):
    def __init__(self):
        self.dados = [ ]

def getFila(self):
        return self.dados

def inserirDado(self,novoValor):
        self.dados.append(novoValor)
```

Inserir valores na fila

```
class Fila(object):
    def init (self):
        self.dados = [ ]
    def getFila(self):
        return self.dados
    def inserirDado(self,novoValor):
        self.dados.append(novoValor)
                                          print(filaTeste.getFila())
```

fila.py

programa.py

```
import fila
filaTeste = fila.Fila()
print(filaTeste.getFila())
filaTeste.inserirDado(2)
```

Exercício C

Crie uma classe chamada Fila. Em seguida, insira os seguintes valores na lista: "IFPB", "ED", "2019.2", "João Pessoa", "Estágio 1". Mostre os valores da lista antes e depois da inserção.

Operações

- Criar fila
- Mostrar valores da fila
- Inserir valores
- Remover valores
- Definir tamanho da fila

"primeiro" "segun	o" "terceiro"	"quarta"	"quinta"	
-------------------	---------------	----------	----------	--





fila.py

```
class Fila(object):
    def __init__(self):
        self.dados = [ ]

    def getFila(self):
        return self.dados

    def removerDado(self):
        self.dados.pop(0)
```

fila.py

```
class Fila(object):
                                           import fila
    def init (self):
        self.dados = [ ]
                                           filaTeste = fila.Fila()
                                           filaTeste.inserirDado(0)
    def getFila(self):
                                           filaTeste.inserirDado(1)
        return self.dados
                                           filaTeste.inserirDado(2)
                                          print(filaTeste.getFila())
    def removerDado(self):
        self.dados.pop(0)
                                           filaTeste.removerDado()
                                          print(filaTeste.getFila())
```

programa.py

Exercício D

Utilize a Fila criada no exemplo anterior ("IFPB", "ED", "2019.2", "João Pessoa", "Estágio 1") e remova um elemento da fila (aquele que chegou primeiro na fila).

Se quisermos remover um item específico da fila, como implementar?



"primeiro" "segun	o" "terceiro"	"quarta"	"quinta"	
-------------------	---------------	----------	----------	--









Remover valores

fila.py

```
class Fila(object):
    def init (self):
        self.dados = [ ]
    def getFila(self):
        return self.dados
    def remove(self, valor):
        pos = self.dados.index(valor)
        for i in range(0,pos+1):
             self.dados.pop(0)
```

Remover valores

```
class Fila(object):
                                           import fila
    def init (self):
                                          filaTeste = fila.Fila()
        self.dados = [ ]
                                           filaTeste.inserirDado(0)
                                           filaTeste.inserirDado(1)
    def getFila(self):
                                           filaTeste.inserirDado(2)
        return self.dados
                                          print(filaTeste.getFila())
    def remove(self,valor):
                                           filaTeste.removerDado(1)
        pos = self.dados.index(valor)
                                          print(filaTeste.getFila())
        for i in range(0,pos+1):
             self.dados.pop(0)
```

programa.py

fila.py

Exercício E

Utilize a Fila ("IFPB", "ED", "2019.2", "João Pessoa", "Estágio 1") e remova o elemento "2019.2" da fila.

OBS: para que 2019.2 seja eliminado todos os valores que chegaram antes desse elemento também precisa ser removido.

Operações

- Criar fila
- Mostrar valores da fila
- Inserir valores
- Remover valores
- Definir tamanho da fila

Definir tamanho da fila

fila.py

```
class Fila(object):
    def init (self):
        self.dados = [ ]
    def getFila(self):
        return self.dados
    def tamanhoFila(self):
        return len(self.dados)
```

Definir tamanho da fila

```
fila.py
class Fila(object):
    def init (self):
        self.dados = [ ]
    def getFila(self):
        return self.dados
    def tamanhoFila(self):
         return len(self.dados)
```

programa.py

```
import fila
filaTeste = fila.Fila()
print(filaTeste.getFila())
filaTeste.inserirDado(1)
filaTeste.inserirDado(2)
filaTeste.inserirDado(3)
print(filaTeste.getFila())
print(filaTeste.tamanhoFila())
```

Exercício F

Informe o tamanho da fila ("IFPB", "ED", "2019.2", "João Pessoa", "Estágio 2")

Dúvidas?



alanamm.prof@gmail.com