

## A. EAD Pilha

Considere as seguintes definições dos tipos de dados **DadosPilha**, **NodoPilha** e **PNodoPilha** (apontamentos das aulas teóricas):

```
typedef struct {
    int numAluno;           // número de aluno (chave): { 70000, ..., 75000 }
    float notasMTP[2];     // notas dos mini-testes práticos: [0.0, 2.0]
    float notasTE[2];     // notas dos testes escritos: [0.0, 8.0]
    int notaFinal;        // nota final (valor arredondado da soma das 4 notas): { 0, ..., 20 }
} DadosPilha;

struct NodoPilha {
    DadosPilha Elemento;
    struct NodoPilha *Ant;
};

typedef struct NodoPilha *PNodoPilha;
```

Cada elemento (registo) do tipo **DadosPilha** corresponde aos dados sobre a avaliação de um aluno à disciplina de AED, durante a época de Aprendizagem. O campo **numAluno** é a chave (único).

Descarregar as seguintes bibliotecas (**Folhas práticas ---> Bibliotecas e Exercícios das folhas práticas ---> EAD Pilha (grupo A)**):

**Aleatorio.h** ---> com as operações para gerar números aleatoriamente  
**EADPilha.h** ---> com os tipos e as operações básicas sobre **Pilhas**  
**OperacoesBasicasGrupoA.h** ---> com as operações básicas sobre a estrutura **DadosPilha**  
**OutrasOperacoesGrupoA.h** ---> com as operações relativas aos exercícios do grupo A  
**MainGrupoA.c** ---> com o programa principal do grupo A

Elaborar um programa em C que utilize as operações contidas nas bibliotecas referidas e resolva as questões colocadas a seguir, acrescentado-as uma a uma ao programa.

### 1. Implementar uma função

- receba dois inteiros positivos, **A** e **B**,
- crie e devolva uma pilha de elementos do tipo **DadosPilha**, gerados aleatoriamente (usar o gerador de números - biblioteca **Aleatorio**), cujo tamanho é um valor entre **A** e **B**.

Usar esta função para construir uma Pilha **S** com elementos do tipo **DadosPilha**, cujo tamanho é um valor entre 0 e 15. Mostre a Pilha **S**.

### 2. Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- devolva o segundo elemento a partir do topo da Pilha **S**.

Mostre o resultado da operação e a Pilha **S** após a operação realizada.

**3.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- devolva o segundo elemento a partir do topo da Pilha **S**, mas deixando a Pilha **S** inalterada.

Mostre o resultado da operação e a Pilha **S após** a operação realizada.

**4.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha** e um inteiro positivo ( $> 0$ ) **N**,
- devolva o **N-ésimo** elemento a partir do topo da Pilha **S**.

Mostre o resultado da operação e a Pilha **S após** a operação realizada.

**5.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha** e um inteiro positivo ( $> 0$ ) **N**,
- devolva o **N-ésimo** elemento a partir do topo da Pilha **S**, deixando a Pilha **S** inalterada.  
(Sugestão: usar uma Pilha auxiliar).

Mostre o resultado da operação e a Pilha **S após** a operação realizada.

**6.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- devolva o elemento do fundo da Pilha **S**.

Mostre o resultado da operação e a Pilha **S após** a operação realizada.

**7.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- devolva o elemento do fundo da Pilha, deixando a Pilha **S** inalterada.

Mostre o resultado da operação e a Pilha **S após** a operação realizada.

**8.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- devolva o terceiro elemento a partir do fundo da Pilha **S**.

Mostre o resultado da operação e a Pilha **S após** a operação realizada.

**9.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- troque de posição o elemento que está no topo da Pilha **S** com o que está no fundo da Pilha **S** (usar apenas uma pilha auxiliar).

Mostre a Pilha **S após** a operação realizada.

**10.** Implementar uma função que

- receba uma Pilha **S** com elementos do tipo **DadosPilha**,
- coloque junto ao fundo da Pilha **S** os elementos de **S** com valor no campo **notaFinal** menor do que 10, e junto ao topo da Pilha **S** os elementos de **S** com valor no campo **notaFinal** maior ou igual a 10 (usar duas Pilha auxiliares).

Mostre a Pilha **S após** a realização da operação pedida.

## B. EAD Fila

Considere as seguintes definições dos tipos de dados **DadosFila**, **NodoFila** e **PNodoFila** (apontamentos das aulas teóricas):

```
typedef struct {
    int numAluno;           // número de aluno: { 70000, ..., 75000 }
    float notasMTP[2];     // notas dos mini-testes práticos: [0.0, 2.0]
    float notasTE[2];     // notas dos testes escritos: [0.0, 8.0]
    int notaFinal;        // nota final (valor arredondado da soma das 4 notas): { 0, ..., 20 }
} DadosFila;

struct NodoFila {
    DadosFila Elemento;
    struct NodoFila *Prox;
};

typedef struct NodoFila *PNodoFila;
```

Cada elemento (registo) do tipo **DadosFila** corresponde aos dados sobre a avaliação de um aluno à disciplina de AED, durante a época de Aprendizagem.

Copiar as seguintes bibliotecas:

**Aleatorio.h** ---> com as operações para gerar números aleatoriamente

**EADFila.h** --> com os tipos e as operações básicas sobre **Filas**

**OperacoesBasicasGrupoB.h** --> com as operações básicas sobre a estrutura **DadosFila**

**OutrasOperacoesGrupoB.h** --> com as operações a implementar do exercício B

**MainFilasExB.c** ---> com o programa principal dos exercícios do grupo B

que se encontram disponíveis na página web da disciplina (**Folhas práticas** ---> **Bibliotecas e Exercícios das folhas práticas** ---> **EAD Fila (grupo B)**).

Elaborar um programa em C que utilize as operações contidas nas bibliotecas referidas e resolva as questões colocadas a seguir, acrescentado-as uma a uma ao programa.

### 1. Implementar uma função que

- receba dois inteiros positivos, **A** e **B**,
- crie e devolva uma fila de elementos do tipo **DadosFila**, gerados aleatoriamente (usar o gerador de números - biblioteca **Aleatorio**), cujo tamanho é um valor entre **A** e **B**.

Usar esta função para construir uma Fila **Q** com elementos do tipo **DadosFila**, cujo tamanho é um valor entre 0 e 15. Mostre a Fila **Q**.

### 2. Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- devolva o segundo elemento a ser removido da Fila **Q**.

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**3.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- devolva o segundo elemento da frente da Fila **Q**, mas deixando a Fila **Q** inalterada.

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**4.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila** e um inteiro positivo ( $> 0$ ) **N**,
- devolva o **N-ésimo** elemento a ser removido da Fila **Q**.

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**5.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila** e um inteiro positivo ( $> 0$ ) **N**,
- devolva o **N-ésimo** elemento a ser removido da Fila **Q**, deixando a Fila **Q** inalterada (sugestão: usar uma Fila auxiliar).

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**6.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- devolva o último elemento a ser removido da Fila **Q**.

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**7.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- devolva o último elemento a ser removido da Fila **Q**, deixando a Fila inalterada.

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**8.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- devolva o terceiro elemento a partir da cauda da Fila **Q** a ser removido (antepenúltimo elemento da Fila).

Mostre o resultado da operação e a Fila **Q após** a operação realizada.

**9.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- troque de posição o elemento que está na frente da Fila **Q** com o que está na cauda da Fila **Q** (sugestão: usar apenas uma outra Fila auxiliar).

Mostre a Fila **Q antes** e **após** a realização da operação pedida.

**10.** Implementar uma função que

- receba uma Fila **Q** com elementos do tipo **DadosFila**,
- coloque junto à cauda da Fila **Q** os elementos com valor no campo **notaFinal** menor que 10, e junto à frente da Fila **Q** os elementos com valor no campo **notaFinal** maior ou igual a 10 (sugestão: usar duas Filas auxiliares).

Mostre a Fila **Q após** a realização da operação pedida.

## C. EAD Pilha e EAD Fila

Considere as definições de tipos de dados dos exercícios A e B: **DadosPilha**, **NodoPilha**, **PNodoPilha**, **DadosFila**, **NodoFila**, **PNodoFila** (e apontamentos das aulas teóricas), assim como as bibliotecas que lhes estão associadas que se encontram disponíveis na página web da disciplina (**Folhas práticas ---> Bibliotecas e Exercícios das folhas práticas ---> EAD Pilha + EAD Fila (grupo C)**).

1. Elaborar um programa em C que

- construa uma Pilha **S** com elementos do tipo **DadosPilha**, cujo tamanho é um valor entre 0 e 15, usando a operação **criarPilhaAleatoria**
- mostre a Pilha **S**

2. Implementar uma função que

- receba uma Pilha **S**,
- inverta a ordem dos elementos da Pilha **S**, utilizando apenas uma estrutura auxiliar.

Definir adequadamente a estrutura auxiliar e prever a possibilidade da Pilha estar vazia.

Acrescentar ao programa elaborado em 1 os seguintes passos:

- inverta a ordem dos elementos na Pilha **S**, usando a função implementada
- mostre a Pilha **S** invertida

## D. EAD Fila e EAD Pilha

Considere as definições de tipos de dados dos exercícios A e B: **DadosPilha**, **NodoPilha**, **PNodoPilha**, **DadosFila**, **NodoFila**, **PNodoFila** (e apontamentos das aulas teóricas), assim como as bibliotecas que lhes estão associadas, que se encontram disponíveis na página web da disciplina (**Folhas práticas ---> Bibliotecas e Exercícios das folhas práticas ---> EAD Fila + EAD Pilha (grupo D)**).

1. Elaborar um programa em C que

- construa uma Fila **Q** com elementos do tipo **DadosFila**, cujo tamanho é um valor entre 0 e 15, usando a operação **criarFilaAleatoria**
- mostre a Fila **Q**

2. Implementar uma função que

- receba uma Fila **Q**,
- inverta a ordem dos elementos da Fila **Q**, utilizando apenas uma estrutura auxiliar.

Definir adequadamente a estrutura auxiliar e prever a possibilidade da Fila estar vazia.

Acrescentar ao programa elaborado em 1 os seguintes passos:

- inverta a ordem dos elementos na Fila **Q**, usando a função implementada
- mostre a Fila **Q** invertida