

## TRABALHO PRÁTICO - IMPLEMENTAÇÃO DE BIBLIOTECA DE PERSISTÊNCIA E SISTEMA COM 3 ENTIDADES RELACIONADAS EM LINGUAGEM C

Este trabalho prático engloba dois tópicos distintos:

- 1) a implementação de uma biblioteca de persistência de dados em arquivo binário em linguagem C, e
- 2) a implementação de um sistema com 3 entidades relacionadas, como por exemplo, Clientes, Produtos e Vendas. O sistema deve usar a biblioteca de persistência para gravar e recuperar as informações gravadas em disco.

### 1) Biblioteca de Persistência

Este tópico do trabalho refere-se à implementação da biblioteca de persistência de dados, cujo objetivo é centralizar as operações mais comuns de gravação de dados em arquivos, tais como o *CRUD* (*Create, Retrieve, Update, Delete*).

Cada arquivo será representado por um descritor com a seguinte estrutura:

```
struct dFile {  
    FILE* arquivo;           // referência ao arquivo em disco  
    int tamanhoRegistro;    // quantidade de bytes do tipo de dado (struct) a ser gravado/lido  
};
```

Os tipos de dados e as operações da biblioteca de persistência estão descritos no quadro a seguir.

```

typedef struct dFile DFile;
typedef DFile*      pDFile;

typedef int  (*FuncaoComparacao)(void *, void *);
typedef void* (*FuncaoAloca)      ( );
typedef void (*FuncaoImpressao) (void *);
typedef int  (*FuncaoPredicado)  (void *);

pDFile open  (char[30], int);
void createe (pDFile, void*);           // dados
void* retrieve (pDFile, void*, FuncaoComparacao); // chave
void update  (pDFile, void*, void*, FuncaoComparacao, FuncaoAloca); // chave e os
dados
void delete  (pDFile, void*, FuncaoComparacao); // chave
void close   (pDFile);

pDLista queryAll (pDFile, FuncaoAloca);
pDLista queryBy  (pDFile, FuncaoPredicado);
void   persistAll (pDFile, pDLista);

```

## 2) Sistema com 3 entidades relacionadas

O segundo tópico deste trabalho trata da implementação de um sistema contendo, *no mínimo*, 3 entidades relacionadas, conforme o exemplo de diagrama ilustrado na figura a seguir.

