



TIPOS ABSTRADOS DE DADOS: CONTEXTO E APLICAÇÃO

Curso: Engenharia de Produção
Disciplina: Algoritmos e Estruturas de Dados I
Professor: Alexandre Magno de Sousa

2º semestre de 2018
Data: 17/08/18

1 Contexto

Uma biblioteca precisa realizar o registro do empréstimo dos livros que atualmente é realizado de forma manual e por meio de fichas. Para tanto, precisa de um sistema que armazene os dados dos livros existentes no acervo, os dados dos usuários da biblioteca (estudantes universitários, professores e técnicos administrativos) e os dados referentes aos empréstimos de cada livro.

Para o registro dos livros, os seguintes dados são necessários: ISBN (código do livro), título do livro, editora, local de publicação, data de publicação, primeiro e segundo autor, número da edição e quantidade de livros existentes, e quantidade de livros emprestados. Um livro somente pode ser emprestado se o número de livros disponíveis para empréstimo é maior do que 10% do número total de livros existentes na biblioteca, caso contrário, o livro não poderá ser emprestado, pois deverá permanecer na biblioteca para que os usuários interessados em utilizá-lo possam consultá-lo dentro das instalações da biblioteca.

As informações referentes aos usuários que precisam ser armazenados no sistema para segurança do acervo são: nome e sobrenome, CPF, carteira de identidade, endereço completo (i.e. logradouro, número, complemento, bairro, cidade e CEP), data de nascimento, tipo de usuário (i.e. aluno, professor ou funcionário). Caso o usuário seja um aluno, o nome do curso deverá ser registrado. Se o usuário for um professor ou um funcionário da universidade, o nome do departamento ao qual está vinculado deverá ser registrado.

Para realização de um empréstimo, o usuário deverá apresentar sua carteirinha para o atendente ou bibliotecário. Em seguida, deverá apresentar o(s) livro(s) que deseja levar. Desse modo, o empréstimo deverá registrar o CPF do usuário e o ISBN de cada livro do empréstimo. Pelas regras da biblioteca, o número máximo de livros que poderá ser levado por um usuário são 3 (três) livros. Porém, caso o usuário seja um professor, poderá levar até 5 (cinco) livros. Assim, o sistema registrará a data do empréstimo, o prazo para empréstimo (i.e. duas semanas) e a data de entrega dos livros. A data de entrega somente é registrada quando o usuário realizar a devolução dos livros. Além disso, o usuário poderá fazer o empréstimo se não houver multas referentes a atrasos de empréstimo. O valor da multa é de R\$2,00 (dois reais) por dia de atraso para cada livro.

Diante desse cenário, para o desenvolvimento do sistema, serão necessários a construção de três módulos: (i) módulo do livro, (ii) módulo do usuário e (iii) módulo de empréstimo. Então, realize a modelagem de dados por meio da abstração do problema real para um sistema de cadastro e controle de empréstimos. Leia o contexto do problema, faça a interpretação dos processos envolvidos no empréstimo de livros da biblioteca para dar início à construção das estruturas de dados para armazenar as todas informações necessárias do problema e das operações referentes a cada processo de empréstimo. Esses são os elementos que constituem um Tipo Abstrato de Dados (T.A.D.).

2 Tipo Abstrato de Dados (T.A.D.)

Um TAD envolve dois elementos essenciais: (a) os dados e (b) as operações sobre esses dados. Os dados são obtidos a partir da abstração do problema real, representam as informações necessárias envolvidas no processo do problema. O recurso da linguagem C utilizado para a construção dos tipos de dados necessários são os registros ou **structs**, também são chamados de tipos de dados heterogêneos.

As operações nada mais são do que “aquilo que o sistema pode fazer” com os dados e, geralmente, o nome de uma operação é representado por um verbo no infinitivo pois indica uma ação realizada sobre os dados (e.g. cadastrar, alterar, excluir ou pesquisar). Em uma linguagem de programação, as operações são representadas por meio de funções ou procedimentos¹. Na linguagem C, funções e procedimentos são representados por meio de um bloco chamado de função.

Para construção das operações, sempre deve-se pensar primeiro em responder as seguintes questões:

¹Funções e procedimentos podem receber como entrada parâmetros com os quais eles podem trabalhar, esse processo é chamado de passagem de parâmetro (que pode ser por valor ou referência). A principal diferença entre função e procedimento é que uma função sempre retorna algum valor, enquanto que o procedimento não retorna um valor depois de realizar a sua operação.



1. O que você pode fazer com um elemento (livro, usuário ou empréstimo)? A partir daqui, as operações básicas podem ser desenvolvidas;
2. O que cada operação precisa, em termos de dados, para ser realizada? Ou seja, quais parâmetros são necessários? Deve-se utilizar passagem de parâmetro por valor ou por referência?

Observação: não será permitido o uso de variáveis globais para o desenvolvimento do programa, pois não é uma boa prática de programação.

2.1 Operações Básicas

- **Inserir** (cadastrar): insere um elemento na estrutura de dados;
- **Pesquisar**: realiza a pesquisa por meio de um ID de um elemento passado como parâmetro no vetor de elementos da estrutura de dados e retorna o índice de onde esse elemento se encontra;
- **Alterar**: recebe como parâmetro o índice de um elemento já pesquisado para realizar a alteração dos dados e recebe um elemento para que seja atualizado, assume-se que esse elemento já vem alterado, ou seja, antes da chamada da função **Alterar** deve-se fazer o uso da função **Pesquisar** e da função auxiliar **Ler** vide seção 2.2, essa realiza a leitura dos novos dados para atualização de um determinado elemento;
- **Excluir**: recebe como parâmetro o índice de um elemento já pesquisado para realizar a exclusão dos dados e, neste caso, para excluir um usuário deverá ser realizado o deslocamento dos elementos dentro do vetor (lista) da respectiva estrutura de dados (livros, usuários ou empréstimo) para reorganizá-lo. Antes da chamada da função **Excluir** deve-se fazer o uso da função **Pesquisar** para identificar o índice do elemento que deve ser excluído.

Observação: lembre-se de se perguntar do que cada função precisa para realizar a operação, ou seja, quais dados são necessários para realizar a operação, e quais deles devem ou não ser por passagem de parâmetro por valor ou referência.

2.2 Operações Auxiliares

- **Ler**: recebe como passagem de parâmetro por referência um elemento (livro, usuário ou empréstimo) e realiza a leitura de todos os dados desse elemento. **Observação:** para leitura de strings a função **fgets** será obrigatória, o uso da função **fflush(stdin)** deverá ser feito antes de qualquer função de leitura, seja **fgets** ou **scanf**, isso evita que o programa salte leituras durante a execução.
- **Imprimir**: recebe como passagem de parâmetro por valor um elemento e realiza a impressão de todos os dados da estrutura desse elemento.