

# Programação Estruturada

Prof<sup>a</sup> Andréa Braga  
Prof<sup>a</sup> Lívia Gregorin

## Projeto Biblioteca

### Informações importantes:

- O projeto é composto por: 2 etapas parciais (20%) + entrega final (80%).
- Para cada entrega parcial é necessário enviar **até o prazo de 1 semana**, o arquivo fonte atual (formato .c ou .cpp) do projeto via CANVAS.
- O código fonte das entregas deverá conter **cabeçalho com nº do grupo, RA e nome do trio**.
- **Manter o nome e tipo das variaveis e os membros das estruturas.**
- Utilizar: **alocação dinâmica, ponteiros para memória e ponteiros para arquivos.**
- A entrega final será feita até no máximo dia **03/12** via CANVAS.
- Projetos cujas cópias/colas forem identificadas não serão pontuados, portanto, a nota referente à Avaliação Final será 0 (**zero**).

### 1<sup>a</sup> etapa - Fazer menu e rotina para:

- Arquivo: **alunos.bin** e Cadastro de ALUNOS contendo uma estrutura com:
  - nome
  - RA
  - quantidade de livros emprestados (máximo 3)
  - quantidade de livros reservados (máximo 1)
  - tabela 0 (irá conter o status e o nº de registro do 1º livro, se for o caso)
  - tabela 1 (irá conter o status e o nº de registro do 2º livro, se for o caso)
  - tabela 2 (irá conter o status e o nº de registro do 3º livro, se for o caso)
  - tabela 3 (irá conter o status e o nº de registro do 4º livro, se for o caso)
    - ❖ as tabelas devem ser **struct**
    - ❖ as tabelas devem ser inicializadas com “L” – livre
- Consulta da situação do aluno – 2 tipos de consulta:
  - Total
    - mostrar o nome, RA e a quantidade de livros emprestados e reservados (se for o caso – mostrar as tabelas)
  - Parcial
    - o acesso deve ser feito através do **RA do aluno**
    - mostrar o nome e a quantidade de livros emprestados e reservados (se for o caso – mostrar as tabelas) para aquele aluno

## Exemplo do Cadastro de Alunos

Nome	RA	Emp	Res	Tabela 0	Tabela 1	Tabela 2	Tabela 3
Ana Paula	200301	1	1	E - 002	R - 003	L -----	L -----
Beto Lopes	200302	1	0	E - 003	L -----	L -----	L -----
Caio Ferreira	200303	0	0	L -----	L -----	L -----	L -----

```
typedef struct aluno{
    char nome[80];
    char RA[7];
    int emprestado; // qtde de livros emprestados - Max=3
    int reservado; // qtde de livros reservados - Max =1
    struct info_livro tabela[4]; //Obs.: (p->tabela+i)->sigla
}aluno; //Obs.: (p->tabela+i)->reg
```

```
struct info_livro{
    char sigla; // [L]ivre, [E]mprestado, [R]eservado
    int reg; // registro do livro
};
```

- Arquivo: **livros.bin** e Cadastro de LIVROS contendo uma estrutura com:
  - nº do registro do LIVRO (gerado automaticamente pelo sistema)
  - título
  - autor
  - status irá conter a sigla de status, o RA do aluno, data da retirada e data de devolução (7 dias após retirada), se for o caso
    - L – livre
    - E – emprestado
    - R – reservado (data da retirada – 1 dia após a devolução do livro)
  - ❖ status deve ser **struct**
  - ❖ o status deve ser inicializado com “L” – livre
  - ❖ cada livro poderá ter apenas 1 empréstimo e 1 reserva
- Consulta de LIVROS:
  - Total

- ❖ mostrar o nº do registro, título, autor e status (se for o caso, mostras as tabelas)
- Parcial (status)
  - ❖ deve-se escolher um determinado **STATUS (L, E ou R)**
  - ❖ mostrar o nº do registro, o autor e status (se for o caso, mostras as tabelas)
- Parcial (título)
  - ❖ a escolha do **título**, deve ser feita pelo usuário
  - ❖ mostrar nº do registro, autor e status (se for o caso – mostrar as tabelas)

## Exemplo do Cadastro de Livros

Reg	Título	Autor	status 0				status 1			
			Sigla	RA	Ret	Dev	Sigla	RA	Ret	Dev
001	Tecle e Aprenda C	Herbet, S.	L	---	---	---	---	---	---	---
002	Algoritmos e Pseudocódigos	Mizan, T.	E	200301	24/05	31/05	---	---	---	---
003	Teoria do Compilador	Silav, C.	E	200302	27/05	03/06	R	200301	04/06	11/06
004	Linguagem C	Herbet, S.	L	---	---	---	---	---	---	---

```

typedef struct livro{
    int             reg;                      // gerado automaticamente
    char            titulo[80];
    char            autor[80];
    struct info_aluno status[2];           //Obs.: (p->status+i)->sigla
}aluno;

} livro;

struct info_aluno{
    char            sigla;
    char            RA[7];
    int             dia_ret;                 // dia - máx=31
    int             mes_ret;                 // mes: fev=28 dias
    int             dia_dev;                 // mes: abril=jun=set=nov=30 dias
    int             mes_dev;                 // os demais = 31 dias
};

```

## 2ª etapa - Fazer menu e rotina para:

- Fazer rotina para **emprestimo - reserva**:
    - o acesso deve ser feito através do **RA**
    - verificar se é aluno permitido, isto é, **quantidade de empréstimos inferior a 3 ou quantidade de reserva inferior a 1**, caso positivo:
      - ❖ entrar com o **título** do livro a ser emprestado/reservado
      - ❖ verificar no **cadastro de livro**:
- se status **L (livre)**, caso positivo, alterar:
- **cadastro de livro** - status do livro para **E + RA + data retirada + data devolução**
  - **cadastro de alunos** – alterar quantidade de empréstimo + a tabela para **E + nº de registro do livro**
- se status **E (emprestado)**, caso positivo, verificar se livro não tem reserva, caso positivo:
- **cadastro de livro** - status do livro para **R + RA + data retirada + data devolução**
  - **cadastro de alunos** – alterar quantidade de reserva + a tabela para **R + nº de registro do livro**
- se status **R (reservado)**, caso positivo:
- mensagem de livro já reservado, mostrar **RA** e data de devolução da reserva
    - ❖ mostrar nº do registro, título, autor e status alterado (mostrar as tabelas)
    - ❖ mostrar o nome, RA e a quantidade de livros emprestados e reservados (mostrar as tabelas)