

## Especificação do Projeto em Linguagem C

O projeto em C será composto pelo sistema descrito neste documento. A entrega será realizada via moodle em um único arquivo. O arquivo deve ser nomeado da seguinte forma:

- Nome.Sobrenome.matricula.ProjetoC.

Um exemplo de arquivo gerado seguindo essas regras seria:

- Fabiana.Mendes.2000000.ProjetoC.

## Sistema de Gerência de Clube Esportivo

Você foi contratado por um clube esportivo para desenvolver um sistema informatizado para gerenciar os associados do clube. Esse sistema será usado pela secretaria e deverá conter um menu com as seguintes opções:

1. Cadastrar associado
2. Cadastrar dependente
3. Listar associados
4. Listar associados por modalidade
5. Pesquisar associados
6. Gerar relatório de associados
7. Finalizar Programa

Cada item de menu deverá ser realizado por um procedimento/função específico. A seguir é apresentada a especificação de cada um dos itens do menu.

**1. Cadastrar associado:** O cadastro do associado deverá envolver **identificador**, **nome**, **idade**, e até três dependentes. Para cada dependente é armazenando o **nome** e a **idade**. Além disso no momento do cadastro é informado se o associado realiza alguma aula das seguintes modalidades: Natação, Futsal, Tênis.

- **Identificador** é um número inteiro
- **Nome** é uma string de no máximo 20 caracteres
- **Idade** é um inteiro positivo. Só deverão ser aceitas idades com valores positivos. Enquanto não for inserida uma idade positiva, seu programa deve continuar apresentando erro pedindo que o usuário insira novamente.

O associado deve ser armazenado em uma struct da seguinte forma:

```
struct Associado {  
    int id;  
    char nome[20];  
    int idade;  
    int qtdDep;  
    Dependente dep[3];  
    bool natacao;  
    bool futsal;  
    bool tenis;  
};
```

**2. Cadastrar dependente:** Para cadastrar um dependente a um associado, é necessário informar antes o identificador do associado. Para cada dependente é armazenando o **nome** e a **idade**. Será possível cadastrar no máximo três dependentes por associado.

- **Nome** é uma string de no máximo 20 caracteres
- **Idade** é um inteiro positivo. Só deverão ser aceitas idades com valores positivos. Enquanto não for inserida uma idade positiva, seu programa deve continuar apresentando erro pedindo que o usuário insira novamente.
- Caso o usuário tente cadastrar mais de três dependentes para um mesmo associado, deve ser informado que "O número máximo de dependentes foi alcançado."
- Após cadastrar o dependente deve ser impressa uma lista com o nome e idade dos dependentes do associado.

O dependente deve ser armazenado em uma struct da seguinte forma:

```
struct Dependente {  
    char nome[20];  
    int idade;  
};
```

**3. Listar associados:** A lista deverá apresentar todos os associados cadastrados no sistema, informando: Identificador - Nome - Idade - Número de dependentes - Se faz Natação - Se faz Futsal - Se faz Tênis. Deverá ser apresentado um associado por linha. **Exemplo de saída:**

- 1 - Daniel Porto - 18 - 2 - Não faz Natação - Faz Fustal - Não Faz Tênis
- 2 - Fabiana Mendes - 19 - 1 - Faz Natação - Não Faz Fustal - Não Faz Tênis
- 3 - Rafael Fazzolino - 20 - 0 - Faz Natação - Faz Fustal - Faz Tênis

**4. Listar associados por modalidade:** O usuário irá selecionar uma das três modalidades(Natação, Futsal, Tênis) para a qual deseja-se listar e o seu programa deve apresentar o identificador e o nome de todos os associados que fazem aula daquela modalidade. **Exemplo de saída:**

- 4 - Mateus
- 8 - Marcos
- 12 - Lucas
- 16 - João

**5. Pesquisar associados:** Seu programa deverá apresentar a opção de inserir o nome do associado que se deseja pesquisar. Então o programa deverá analisar todos os associados para descobrir se o nome buscado está presente.

- Caso o nome buscado esteja na lista, seu programa deverá apresentar como resultado as informações do associado, no seguinte formato: **Identificador - Nome - Idade - Número de dependentes - Quantidade de aulas que faz - Valor da mensalidade**
- O valor da mensalidade é calculado da seguinte forma: R\$100,00 pelo associado; R\$50,00 por cada dependente; R\$30,00 por cada aula realizada. **Exemplo de Saída:** 10 - Fulano - 23 - 2 - 1 - R\$230,00
- No caso de existirem associados com o mesmo nome, todos deverão ser apresentados, um associado por linha.
- Caso o associado não seja encontrado na lista, o seu programa deve exibir a seguinte mensagem de erro: "Associado não encontrado!".

#### **6. Gerar relatório de associados**

Seu programa deve apresentar as seguintes informações sobre os associados cadastrados no sistema:

- Número total de associados cadastrados
- Número total de dependentes cadastrados
- Número de associados que fazem aula de Natação

- Número de associados que fazem aula de Futsal
- Número de associados que fazem aula de Tênis
- Valor total de todas as mensalidades dos associados
- Valor médio das mensalidades dos associados
- Valor de mensalidade mais alto pago por um associado
- Nome do(s) associado(s) que fazem mais aulas

**7. Despedir-se do usuário e encerrar o programa!** O programa deve apresentar uma mensagem de despedida e finalizar o programa.

## Instruções Adicionais

- Comente o seu código, facilitando assim o entendimento de suas partes;
- Respeite as regras de indentação adotadas na disciplina;
- Use nomes de variáveis significativos;
- Faça uso adequado das estruturas de programação estudadas;
- Utilize funções/procedimentos para organizar o seu código. Seu programa deve conter no mínimo sete funções/procedimentos (uma para cada opção do menu).

**ATENÇÃO:** O trabalho deverá ser feito individualmente, não sendo possível que dois ou mais alunos apresentem o mesmo código. Caso seja detectado plágio o/a(s) aluno/a(s) envolvidos/as receberão nota zero no projeto. Você poderá ser convidado para apresentar o seu código.