

Atividades de Fixação - DART - PARADIGMAS DE LINGUAGENS DE PROGRAMAÇÃO

1. Calculadora Simples

Problema: Crie uma calculadora simples que receba dois números e um operador (+, -, *, /).

Desenvolvimento: A aplicação deve permitir a entrada de dois números e um operador, e deve exibir o resultado. A calculadora deve validar os inputs e exibir uma mensagem de erro apropriada se o operador for inválido ou se a divisão for por zero.

2. Conversor de Temperatura

Problema: Desenvolva um programa que converta temperaturas entre Celsius, Fahrenheit e Kelvin.

Desenvolvimento: O programa deve receber a temperatura e a unidade inicial e realizar a conversão para as outras duas unidades. Implemente funções separadas para cada conversão.

3. Validador de CPF

Problema: Implemente um validador de CPF utilizando Dart.

Desenvolvimento: O programa deve receber um CPF no formato 000.000.000-00, verificar sua validade e retornar se é válido ou não. O algoritmo de validação deve seguir as regras brasileiras de verificação de dígitos.

4. Manipulação de Arquivos

Problema: Crie um programa que leia e escreva dados em um arquivo de texto.

Desenvolvimento: O programa deve permitir que o usuário escreva uma série de linhas que serão gravadas em um arquivo de texto. Em seguida, ele deve abrir o arquivo, ler o conteúdo e exibi-lo no console. Valide se o arquivo existe antes de tentar lê-lo.

5. Sistema de Tarefas (TODO List)

Problema: Desenvolva um sistema simples de gerenciamento de tarefas.

Desenvolvimento: O programa deve permitir adicionar, listar, editar e remover tarefas. Cada tarefa deve conter um título e um status de concluída ou não. As tarefas devem ser salvas em uma estrutura de dados e exibidas no console.

6. Jogo da Adivinhação

Problema: Crie um jogo onde o computador escolhe um número aleatório entre 1 e 100, e o usuário deve adivinhar o número.

Desenvolvimento: O jogo deve informar se o palpite do usuário é maior ou menor que o número correto até que ele acerte. Adicione um contador de tentativas e exiba o número

total de tentativas no final.

7. Gerador de Números Primos

Problema: Desenvolva um programa que gere todos os números primos até um número n fornecido pelo usuário.

Desenvolvimento: O programa deve receber um número inteiro n e exibir todos os números primos entre 1 e n . Utilize uma função separada para verificar se um número é primo.

8. Agenda de Contatos

Problema: Implemente uma agenda de contatos simples.

Desenvolvimento: O programa deve permitir adicionar, listar, editar e excluir contatos. Cada contato deve possuir um nome, número de telefone e e-mail. Armazene os contatos em uma lista e exiba-os de forma organizada.

9. Fatorial Recursivo

Problema: Crie uma função recursiva em Dart para calcular o fatorial de um número.

Desenvolvimento: O programa deve receber um número inteiro positivo e calcular seu fatorial utilizando uma função recursiva. Exiba o resultado no console.

10. Conversor de Números Romanos

Problema: Desenvolva um conversor que transforme números inteiros em números romanos.

Desenvolvimento: O programa deve receber um número inteiro e converter para sua representação romana. O programa também deve permitir o caminho inverso, convertendo números romanos em inteiros.

Orientações para implementação e entrega:

Está liberado realizar a atividade em equipes, porém o repositório do GITHUB precisa deixar claro quem contribuiu para construção de cada problema. A entrega obrigatoriamente precisa ser feita no github, onde cada atividade precisa estar contida dentro de uma pasta separada no mesmo repositório e cada pasta deve ter um README apresentando o programa. O link do repositório e os integrantes da equipe precisam estar contida na resposta do fórum da atividade no STUDEO