

# Estudo de Caso - Matemática Financeira



## 1. Descrição do problema

Deve-se criar um programa em Javascript, respeitando os aspectos de programação funcional, para realizar um conjunto de cálculos relacionados à **Matemática Financeira** ilustrados em <https://fazaconta.com/calculadora-financeira.htm>.

Para cada função, deve-se **analisar a adequabilidade de definir especializações** e disponibilizá-las para facilitar a vida do usuário. Exemplo: pode-se criar funções chamadas `desconto10`, `desconto20` e `desconto30`, que já calculam os respectivos descontos. Isso significa que você deve-se criar funções da maneira mais genérica possível de modo a fomentar o reuso ao máximo, através da **Aplicação Parcial** e **Curificação**.

## 2. Entrada de dados

A solução deverá permitir a entrada de dados por parte do usuário de seu programa. Isso poderá ser feito através de uma **interface HTML**, por exemplo. Essa interface não precisa ser rebuscada, basta ser organizada com elementos simples de entrada de dados e exibição dos resultados.

### 3. Como apresentar a solução?

A solução deverá contar com ao menos um arquivo *.html* (se optou pela entrada com HTML) e um arquivo *.js*. Entretanto, tem-se liberdade para organizar a solução em quantos arquivos forem necessários.

O(s) arquivo(s) *.js* deve(m) conter os códigos Javascript para resolver os problemas, naturalmente, organizado(s) em funções que devem vir acompanhadas de comentários sobre seus respectivos funcionamentos.

### 4. Restrições

- NÃO se deve fazer uso de funções pré-definidas em JavaScript que resolvam o problema diretamente; a LÓGICA de programação deve ser toda implementada de modo a viabilizar análise do aprendizado.
- NÃO se deve fazer uso de construções explicitamente relacionadas a um paradigma de programação que não o funcional. Nesse sentido, o uso de laços de repetição como `for`, `while`, `forEach` e equivalentes ou definição de entidades que não sejam exclusivamente constantes (conforme sempre visto no curso) em hipótese alguma deve ser utilizado.

### 5. Uma solução disponível

O arquivo disponível para download a seguir é uma solução criada por um aluno de uma turma da disciplina de Programação Funcional do DCOMP/UFS. O arquivo compactado contém o arquivo *.html* e o arquivo *.js*.

Recomenda-se tentar resolver o problema antes de visitar essa solução. Ainda que tome-se conhecimento da solução disponível, recomenda-se realizar modificações e testes nas variadas funções para experimentar e treinar alternativas interessantes.

**SOLUÇÃO** 