FUNDAMENTOS DA COMPUTAÇÃO

Abstração


Interface
Abstração
Remoção de detalhes
e foco no essencial
Generalização

Abstração: é a chave para o controle da complexidade

Complexidade x Abstração:

- **Resolver um problema é fácil se o problema é pequeno e pouco complexo**
- **Quando o problema é grande e difícil, a complexidade é o inimigo**
- **A abstração é a ferramenta para vencer a complexidade**

Aprender a escolher (e justificar) a ferramenta mais apropriada de abstração é fundamental na computação!

Em todo seu curso de computação você desenvolverá essa habilidade.

Abstração: tem 2 "dimensões"

A abstração tem 2 grandes "dimensões":

- Processo de remoção de detalhes e foco no essencial

- Processo de generalização

Abstração: remoção de detalhes e foco no essencial

Uma abstração correta nos permite remover (ou esconder) os detalhes não importantes no momento, para podermos nos focar no que é realmente essencial.

Veloce Cockpit, (https://www.velocecockpit.com.br/)

Ex.: esconder detalhes dos usuários:

- Qual a abstração para dirigirmos um carro?

- Essa abstração nos fornece qual interface?

*** chave, volante, acelerador, freio, embreagem...**

- Mesmo que a tecnologia de implementação da abstração mude, a abstração em si (e a interface) não muda:

*** injeção eletrônica, carro elétrico, ABS, ...**

- Colocar mais detalhes do que o necessário pode ser ruim:

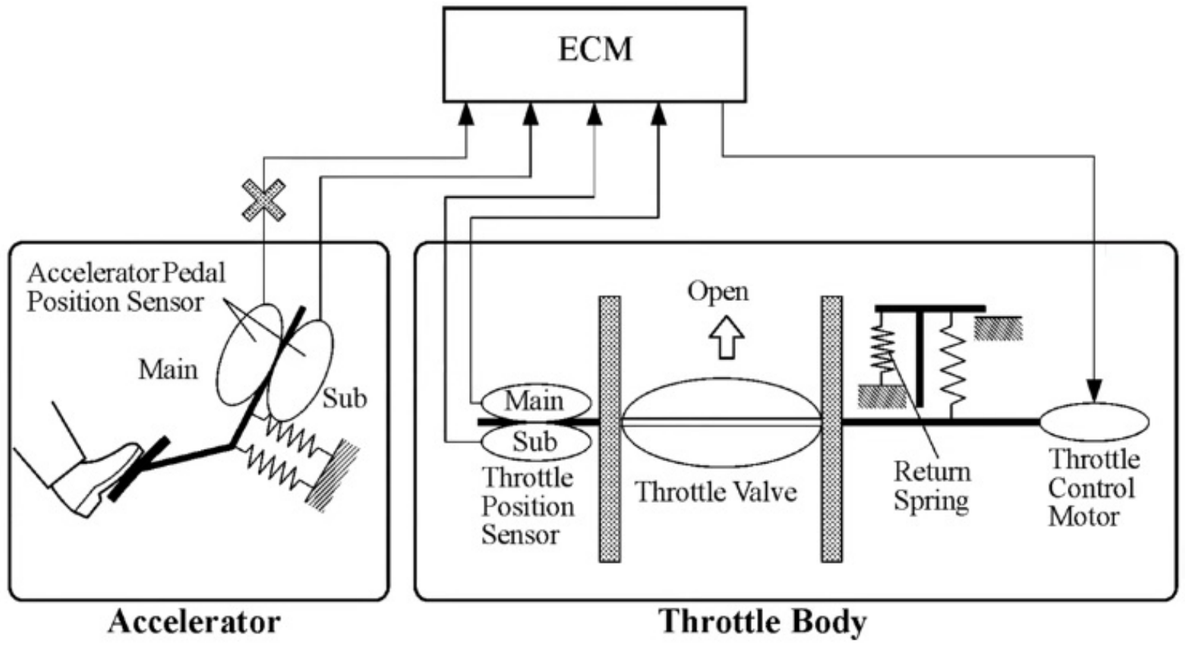
*** controle para ABS?**

Abstração: remoção de detalhes e foco no essencial

Uma abstração correta nos permite remover (ou esconder) os detalhes não importantes no momento, para podermos nos focar no que é realmente essencial.

Ex.: esconder detalhes de outros projetistas:

- O projetista do "Engine Control Module (ECM)" não precisa saber como a "Return Spring" funciona.

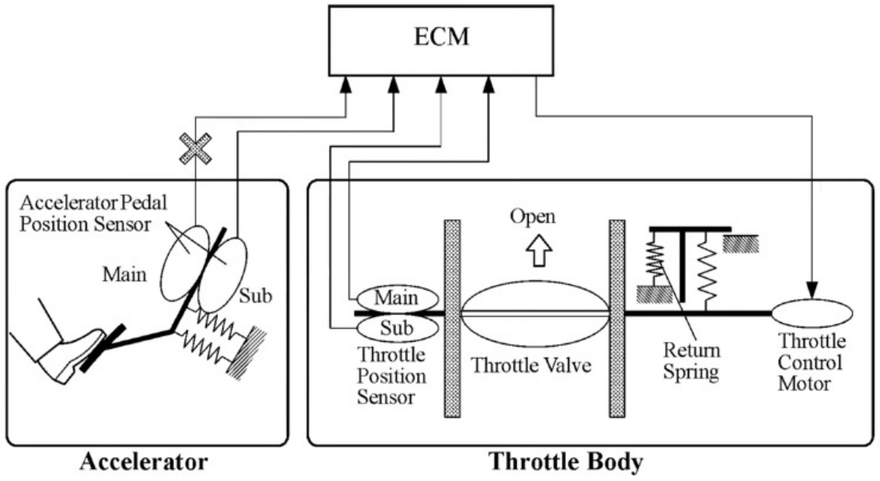


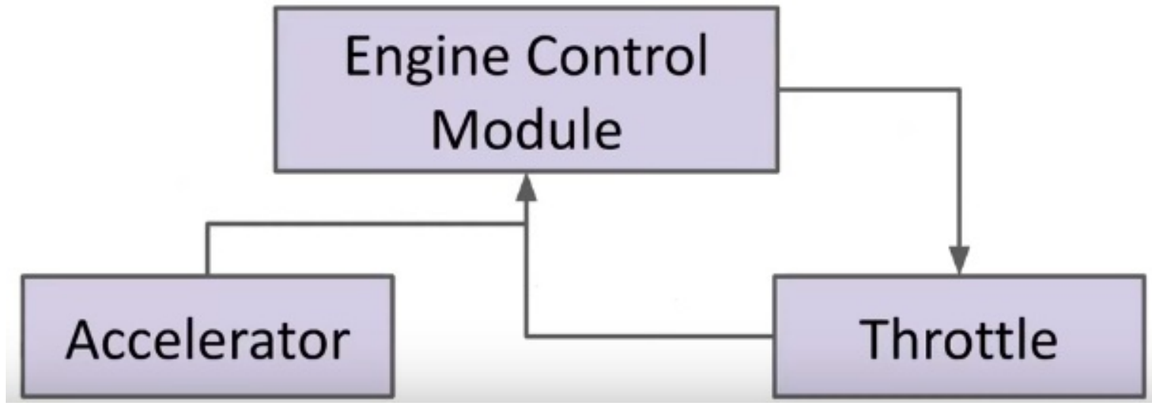
Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/sp22/)

Abstração: remoção de detalhes e foco no essencial

A abstração facilita a decomposição em uma hierarquia de módulos com funcionalidade precisa, interface definida e reusáveis! Esses módulos abstraem/escondem a complexidade interna e permitem que outros os utilizem sem se preocupar com os detalhes internos.

OpenClipart-Vectors, no Pixabay
(https://pixabay.com/vectors/cardboard-box-box-cardboard-package-155479/)





Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/sp22/)

Abstração: generalização

Uma abstração correta nos permite generalizar a solução para situações semelhantes, formulando conceitos gerais e extraindo características comuns, diminuindo o trabalho repetitivo.

Ex.: mangueira de chuveiro

- pode ser utilizada em quase todos os chuveiros

- não é preciso criar uma nova para cada chuveiro

Magazine Luiza
(https://www.magazineluiza.com.br/chuveirinho-com-mangueira-para-chuveiro-3-metros-ki-util/p/ggak02b79c/cj/chcd/)

Abstração: esconder detalhes e generalizar soluções

Esconder detalhes e focar no essencial:

- O processo de ignorar uma ou mais propriedades de um objeto complexo para se focar no que é essencial.

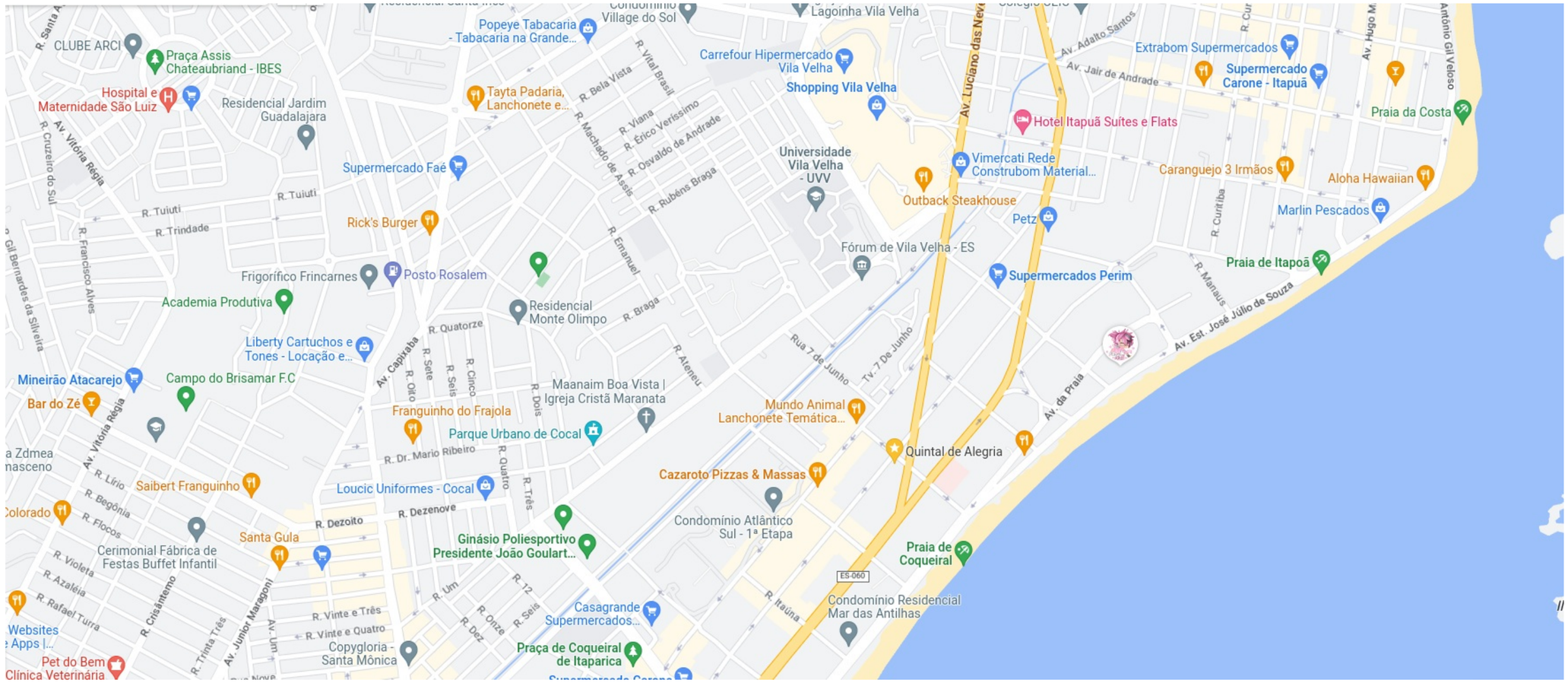
Generalização:

- O processo de formular conceitos gerais, abstraindo-se as propriedades, características e funcionalidades que são comuns.

Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/fa22/)

Abstração: remoção de detalhes

Abstração: remoção de detalhes



Abstração: remoção de detalhes

Abstração: generalização

Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/fa22/)

Para alimentar o cachorro, coloque a comida do cachorro no prato do cachorro.

Para alimentar a galinha, coloque a comida da galinha no prato da galinha.

Para alimentar o coelho, coloque a comida do coelho no prato do coelho.

Etc.

Para alimentar o <animal>, coloque a comida do <animal> no prato do <animal>.

Abstração: generalização

Criar funções é um ótimo exemplo de remoção de detalhes e de generalização!

Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/fa22/)

Uma vez que a função está criada, quem usa não precisa se preocupar com os detalhes internos de implementação, e a função pode ser utilizada e reutilizada diversas vezes.

Abstração: chave para o controle da complexidade

Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/fa22/)

Acima da abstração: só precisamos nos preocupar com a interface, ou especificação, ou contrato. NÃO precisamos nos preocupar COMO (ou POR QUEM) a abstração foi construída.

Barreira da Abstração (Interface)
(interface, especificação, contrato)

Abaixo da abstração: aqui é onde, como, quando e por quem a abstração foi construída. Essa implementação deve ter sido feita de acordo com a interface, especificação ou contrato.

Abstração: chave para o controle da complexidade

Sem a abstração a humanidade não teria construído algumas de suas obras mais importantes.

Tem 2 dimensões:

- **remoção de detalhes e foco no essencial**
- **generalização**

Berkeley CS10: The Beauty and Joy of Computing (https://cs10.org/fa22/)

Quem aprendeu a dirigir um carro em 1930, consegue dirigir outro hoje em dia. A tecnologia de implementação mudou completamente, mas a ABSTRAÇÃO continuou a mesma!

Em resumo:


Interface
Abstração
Remoção de detalhes
e foco no essencial
Generalização