Decomposição **>** Reconhecimento de padrões **>** Abstração **>** Design de algoritmos

Raciocínio lógico

Indução: Leis, Teorias – Fenômeno observado.

Dedução: Leis, Teorias – Previsões e explicações.

Abdução: Diagnósticos, Processo investigativo – Premissa. Ex: “ *A grama está molhada, logo deve ter chovido.*” (talvez sejá).

Inferência:

Sintética: Conclusão experimentação

Abdução e Indução.

Analítica: Dedução e exatas.

Dedução.

Aperfeiçoamento:

**Ato de aperfeiçoar**

* Encontrar solução eficiente;
* Otimizar processos;
* Simplificar linhas de códigos;
* Funções bem definidas.

Decomposição:

* Identificar
* Determinar as etapas
* Executar cada etapa
* Agregar tudo ao finalizar

Ex: Aplicativo

* Finalidade
* Interface
* Funcionalidades **Definição de componentes**
* Pré-requisitos

Padrões:

* Modelo
* Estrutura invariante **Similaridades e Diferenças**
* Repetição

Abstração:

Generalização:

Algoritmos:

* Receber
* Manipular
* Armazena

Como construir um algoritmo?

1. Compreensão do problema
2. Definição dados de entrada
3. Definir processamento
4. Definir dados de saída
5. Utilizar um método de construção
6. Teste e diagnostico.

Construção:

**Narrativa:** Utilização da linguagem natural, diversas interpretações.

**Fluxograma**: Utilização de símbolos pré-definidos, conhecimento prévio da estrutura e dos símbolos.

**Pseudocódigo**: Portugol

GIT x GITHub

Git: sem interface, apenas CLI

Comandos básicos:

Windows:

* cd – Entrar / sair
* dir – listar diretórios
* mkdir – criar pastas
* del / rmdir – deletar arquivos / deletar diretórios (/s /q)

Linux:

* cd – Entrar / sair
* ls – listar diretórios
* mkdir – criar pastas
* rm / rf – deletar arquivos / deletar diretórios (rm -rf )