



EJA IV – Ensino Fundamental
Qualificação Profissional – Informática Básica
Análise e Lógica da Programação

Definição E Características De Um Algoritmo

Rildo Oliveira



27/02/2024

ROTEIRO DE AULA

OBJETO DO CONHECIMENTO:

Definição E Características De Um Algoritmo

HABILIDADE: (EMIFFTP02) Levantar e testar hipóteses para resolver problemas do cotidiano pessoal, da escola e do trabalho, utilizando procedimentos e linguagens adequados à investigação científica.

OBJETIVOS:

Compreender a definição de um algoritmo;

Conseguir realizar tarefas simples na construção com a algoritmos;

DA TEORIA À PRÁTICA: Uso de imagens, texto e conceitos para um melhor entendimento do tema abordado.

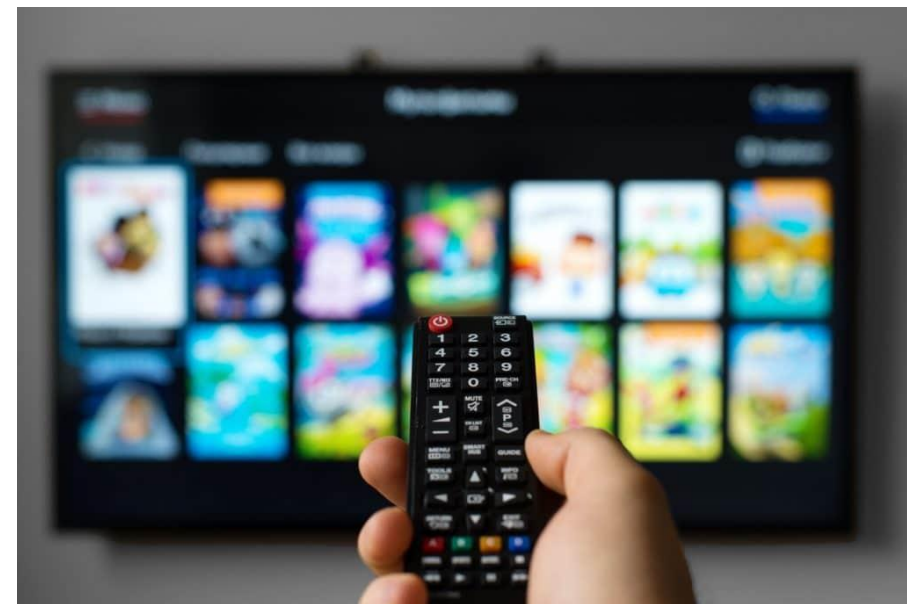
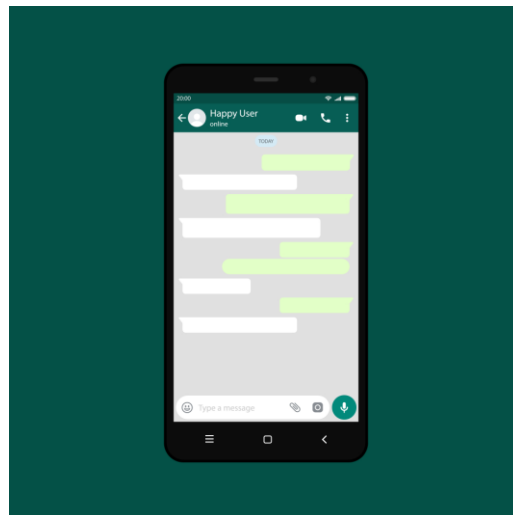
Pasta Compartilhada e Grupo



<https://github.com/rildexter/eja2024/tree/main>



Quais tarefas da sua vida cotidiana são realizadas por algum tipo de computador?

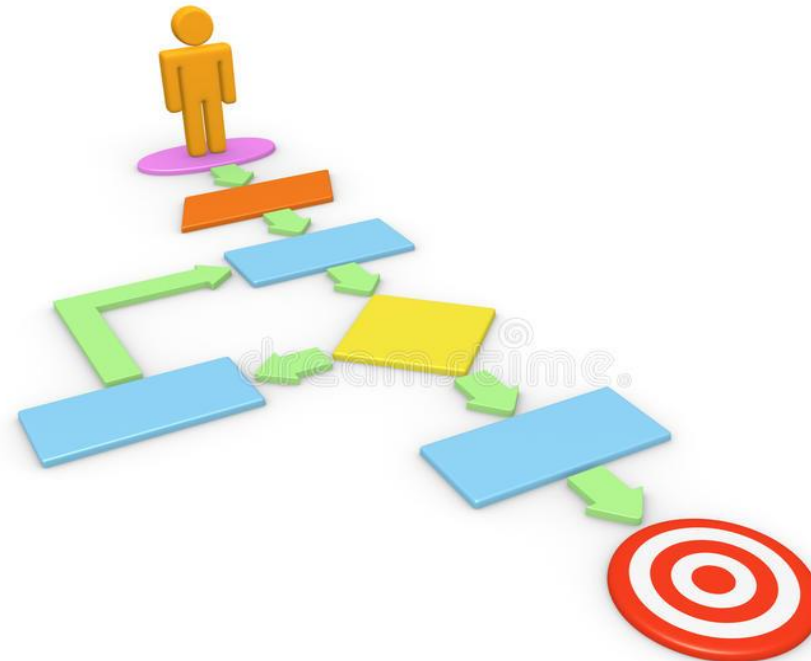


Percepção das tecnologias ao nosso redor

É curioso notar o quanto somos dependentes de uma tecnologia, e normalmente não percebemos a intensidade com que isso acontece. Avanços no processamento e na aquisição de dados (câmeras e outros sensores) permitiram que os computadores fossem aplicados em uma grande quantidade de tarefas.

Algoritmo

Algoritmo é o nome dado à sequência de ações indicando exatamente o que o computador deve fazer para realizar uma tarefa ou resolver um problema.



Desenvolvimento de um Algoritmo

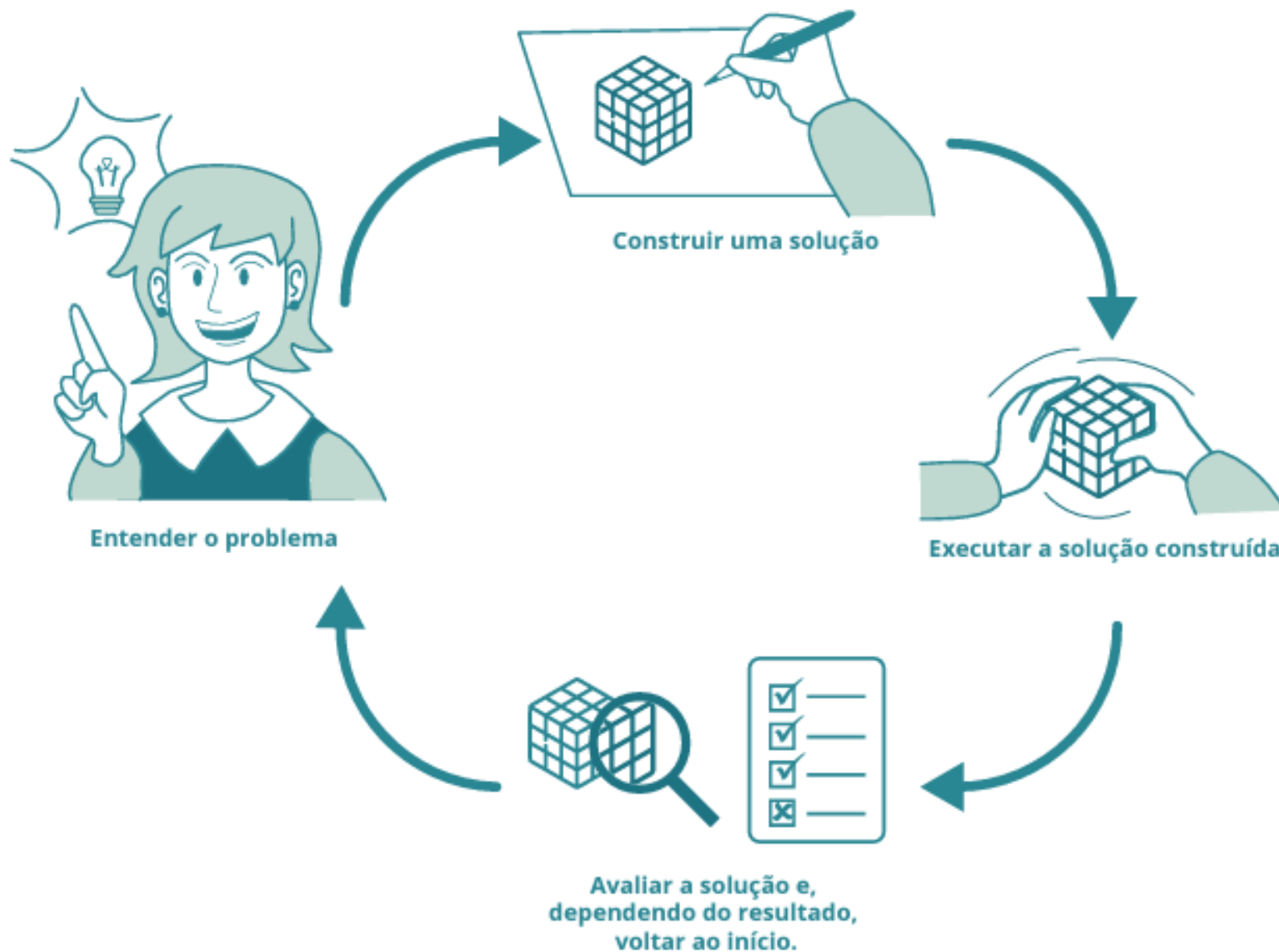
Para o desenvolvimento de um algoritmo eficiente é necessário obedecermos algumas premissas básicas no momento de sua construção:

- Definir ações simples e sem ambiguidade;
- Organizar as ações de forma ordenada;
- Estabelecer as ações dentro de uma sequência finita de passos.

Como projetar um algoritmo?

Escrever um algoritmo é um processo muito parecido com montar um quebra-cabeça.





Torre de Hanoi



Problemática, vamos entender?

1. O quebra-cabeça consiste em uma base contendo três pinos.
2. No primeiro pino, há 3 (três) discos dispostos uns sobre os outros, em ordem crescente de diâmetro, de cima para baixo.
3. O desafio é movimentar todos os discos para o último pino, usando o pino do meio para movimentos auxiliares.
4. Apenas o disco do topo pode ser movimentado.
5. Uma restrição importante deste problema é que um disco nunca pode ficar em cima de outro com diâmetro menor ao dele (KNUTH et al. 1988).

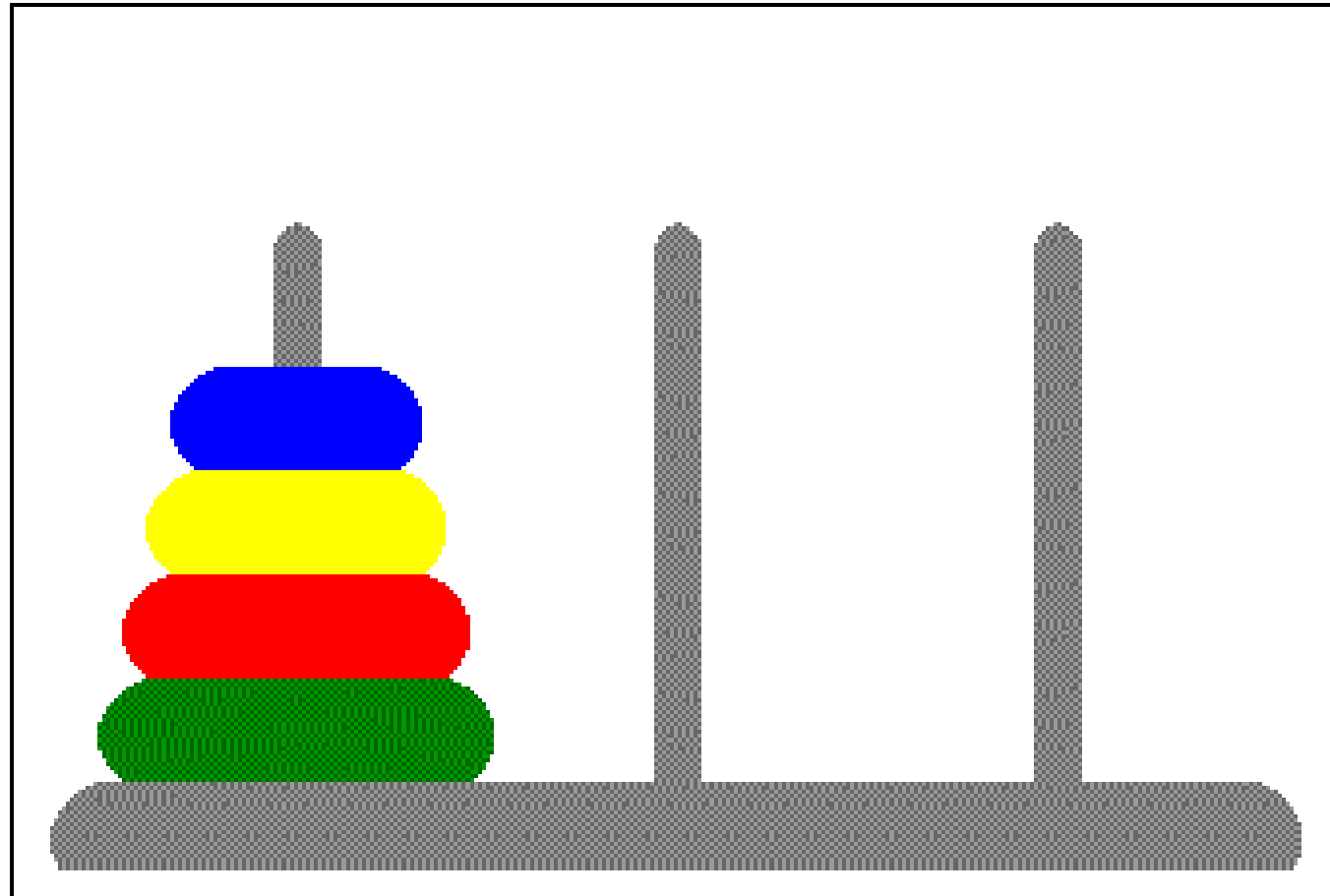


Alcançamos a solução do problema? O que falta para resolver o problema original?



1. Quantas iterações foram necessárias para resolver o desafio anterior ?
1. Quantas interações seriam se fossem 4 discos ?

Quantas interações seriam se fossem 4 discos ?



Exemplo de algoritmo

Algoritmo 1 - Troca de pneu do carro

- 1: desligar o carro;
- 2: pegar as ferramentas (chave e macaco);
- 3: pegar o estepe;
- 4: suspender o carro com o macaco;
- 5: desenroscar os 4 parafusos do pneu furado;
- 6: colocar o estepe;
- 7: enroscar os 4 parafusos;
- 8: baixar o carro com o macaco;
- 9: guardar as ferramentas

Quais são as partes de um algoritmo?

Um algoritmo quando programado num computador é constituído pelo menos das 3 partes, sendo elas:

1. Entrada de dados;
2. Processamento de dados;



Algoritmo 2 - Pegar um ônibus

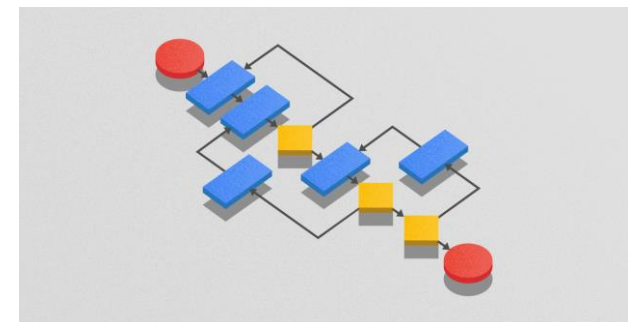
```
1: ir até a parada;  
2: enquanto ônibus não chega  
   faça;  
3:     esperar ônibus  
4: fim-enquanto  
5: subir no ônibus  
6: pegar passagem  
7: se não há passagem então  
8:     pegar dinheiro  
9: fim-se
```

```
10: pagar o cobrador  
11: troco ← dinheiro ← passagem  
12: enquanto banco não está vazio faça  
13:     ir para o próximo  
14: fim-enquanto  
15: sentar
```

Representações de um Algoritmo

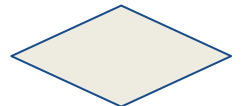
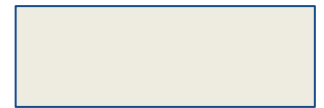
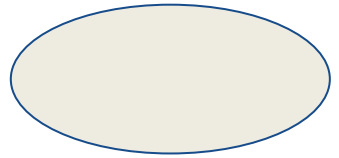
Fluxograma

Os fluxogramas são uma apresentação do algoritmo em formato gráfico. Cada ação ou situação é representada por uma caixa. Tomadas de decisões são indicadas por caixas especiais, possibilitando ao fluxo de ações tomar caminhos distintos.



Representações de um Algoritmo

- O início e o fim do algoritmo são marcados com uma figura elíptica;
- As ações a serem executadas estão em retângulos;
- Sendo que as estruturas de controle condicionais estão em losangos e indicam duas possibilidades de prosseguimento do algoritmo,
 - Uma para o caso da expressão avaliada (condição) ser verdadeira e outra para o caso de ser falsa.



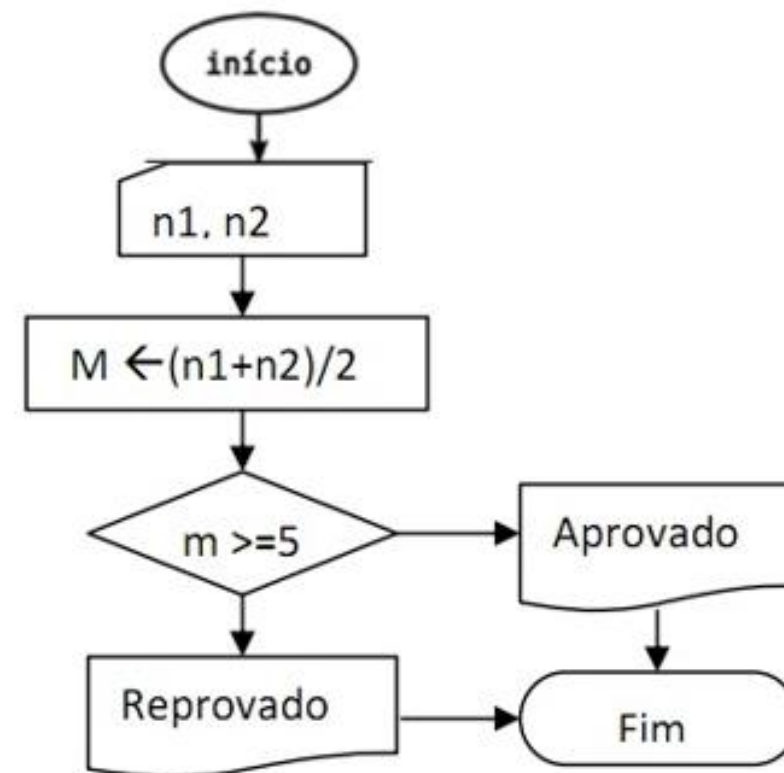
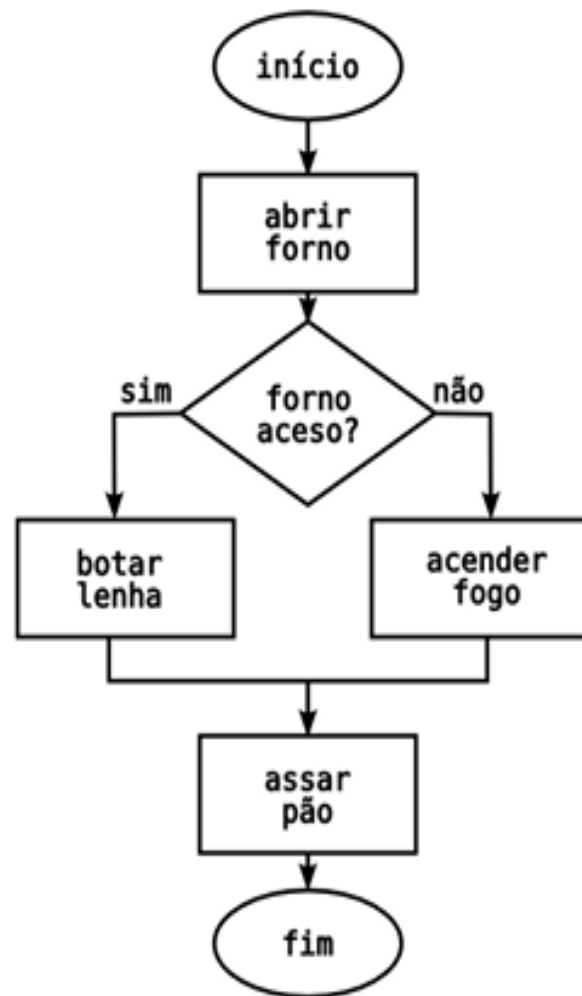


Figura 2.2: Algoritmo representado em forma de um fluxograma.

Programas de Computador - Linguagens

Qualquer tipo de informação que deva ser transferida, processada ou armazenada deve estar na forma de uma linguagem.

- **A linguagem é imprescindível para o processo de comunicação.**

Exemplos

Duas pessoas que se falam o fazem através de uma linguagem em comum, a linguagem natural.

Da mesma forma, duas máquinas trocam informação por uma linguagem, que neste caso mais técnico e restrito, se chama protocolo.



Programas de Computador - Linguagens

Do mesmo modo, um computador armazena suas instruções em código de máquina.

Estas diferentes linguagens não podem ser traduzidas diretamente entre si, pois além de serem **representadas** de **modos diferentes**, também referem-se a coisas muito distintas.

Linguagem de Máquina e Assembler

Esta linguagem é **composta** somente por **números**, representados de **forma binária**, que, sob o ponto de vista do computador, representam as operações e os operandos que serão usados no processamento do programa.

O que essa sequência de códigos representa?

00000000	7F 45 4C 46 01 01 01 00 00 00 00 00 00 00 00 00	.ELF.....
00000010	02 00 03 00 01 00 00 00 D0 82 04 08 34 00 00 004...
00000020	BC 0C 00 00 00 00 00 00 34 00 20 00 07 00 28 004. ... (.

- Qual o resultado você acha que irá gerar com todos esses códigos ?
- Será uma Imagem?
- Um comando para executar um vídeo?

O compilador



Figura 2.3: Compilação: o programa em linguagem de programação é transformado em instruções em linguagem de máquina (que o processador pode executar).

```
00000030  24 00 21 00  06 00 00 00  34 00 00 00  34 80 04 08  ..!.....4...4...
00000040  34 80 04 08  E0 00 00 00  E0 00 00 00  05 00 00 00  4.....
```

Linguagens de Programação

As linguagens de programação são um meio termo entre a linguagem de máquina e a linguagem natural.

As linguagens de programação podem ser classificadas de duas formas:

- Baixo nível;
- Alto nível

Linguagens de baixo nível

As linguagens muito parecidas com linguagem de máquina são chamadas de linguagens de **baixo nível** e suas instruções parecem-se muito com aquelas que serão executadas pelo processador.

```
mov edx,2  
mov esi,4  
add eax,ebx  
sub eax,ecx  
imul edx,eax  
mov eax,edx  
mov edx,0  
cmp esi,0
```

Assembly

Assembler



```
ce ta ed te 07 00 00 00 05 00 00 00 01 00 00 00  
04 00 00 00 38 01 00 00 00 00 00 01 00 00 00  
c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 28 00 00 00 54 01 00 00  
28 00 00 00 07 00 00 00 07 00 00 00 02 00 00 00  
00 00 00 00 5f 5f 74 65 78 74 00 00 00 00 00 00  
00 00 00 00 5f 5f 54 45 58 54 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 1b 00 00 00 54 01 00 00  
00 00 00 00 7c 01 00 00 02 00 00 00 00 04 00 80  
00 00 00 00 00 00 00 00 5f 5f 64 61 74 61 00 00  
00 00 00 00 00 00 00 00 5f 5f 44 41 54 41 00 00  
00 00 00 00 00 00 00 00 1b 00 00 00 0d 00 00 00  
6f 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 24 00 00 00  
10 00 00 00 00 0a 0a 00 00 00 00 00 02 00 00 00  
18 00 00 00 8c 01 00 00 04 00 00 00 bc 01 00 00  
18 00 00 00 0b 00 00 00 50 00 00 00 00 00 00 00  
02 00 00 00 02 00 00 00 01 00 00 00 03 00 00 00  
01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Executável

Linguagens de alto nível

As linguagens de **alto-nível** são as que guardam mais semelhanças com a linguagem natural. Exemplo de linguagens de baixo nível é a linguagem de montagem (assembly).

- **Exemplos de linguagens de alto-nível são:** Pascal, C, Fortran, Java, Perl, Python, Lisp, PHP, entre outras.

Exemplo de programação em “C”

```
main.c
1
2 #include <stdio.h>
3 int main(){
4     printf("Olá Mundo\n");
5 }
6
```

Olá Mundo

...Program finished with exit code 0
Press ENTER to exit console.

Referências

- CORMEN, Thomas H. **Desmistificando Algoritmos**. Rio de Janeiro: Elsevier, 2014.
- EDMONDS, Jeff. **Como pensar sobre algoritmos**. LTC, 2010.
- KLEINBERG, Jon e TARDOS, Éva. **Algorithm Design**. Pearson, 2005.



ATÉ A PRÓXIMA AULA!