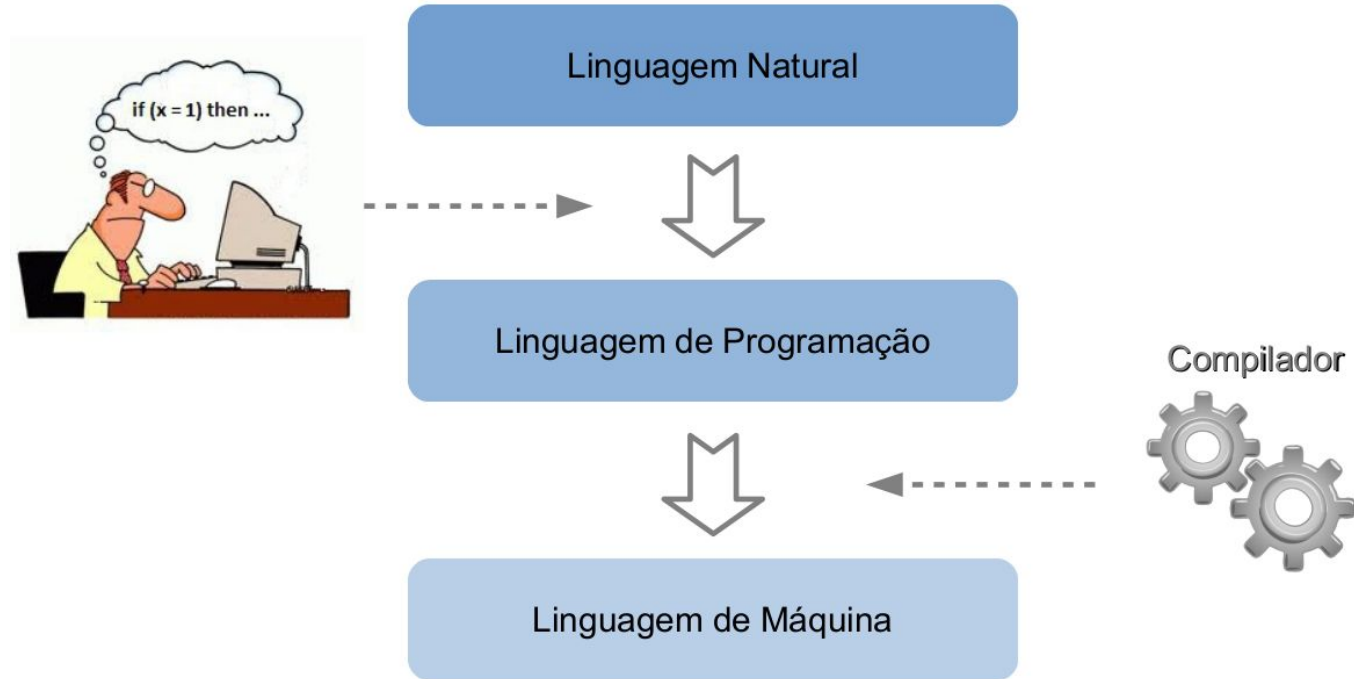




Ambiente e tipos de algoritmos
Profª Simone Dominico

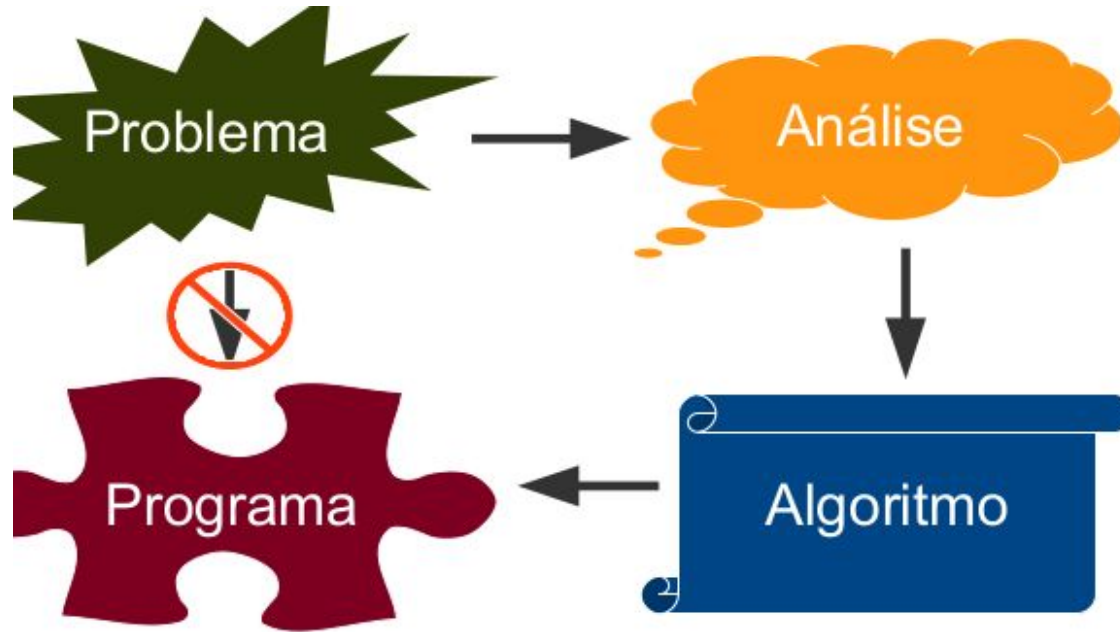
ALGORITMOS



ALGORITMOS

- Paradigma: Procedural ou Imperativo
 - Solução implementada através de ações, executadas sequencialmente
 - Conceitos: variáveis, atribuição, sequenciação
- Baseada nos princípios da Programação Estruturada
- Linguagem de programação que usaremos: Python

SOLUÇÃO DE PROBLEMAS



SOLUÇÃO DE PROBLEMAS

Parte Teórica

1. Ler atentamente o enunciado, visando a sua compreensão.
2. Analisar detalhadamente:
 1. Que resultados devem ser obtidos;
 2. Que dados são fornecidos;
 3. Que procedimentos devem ser realizados para a solução do problema.
3. Construir o algoritmo
4. Simular a sua execução ("teste de mesa").
5. Traduzir o algoritmo em um programa Python

1. Compilar, executar e testar exaustivamente o programa, modificando até que os resultados obtidos correspondam com o previsto.

Parte Prática

ALGORITMOS

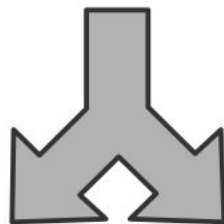
- Propriedades:
 - possui um estado inicial;
 - contém uma sequência lógica e finita de ações (comandos), claras e precisas, com fluxo de execução baseado em:
 - sequência;
 - seleção condicional (seleção de ações);
 - iteração (repetição de ações);
 - possui dados de entrada;
 - produz dados de saída corretos;
 - possui estado final previsível;
 - deve ser eficaz.

ALGORITMOS

- Propriedades:
 - possui um estado inicial;
 - contém uma sequência lógica e finita de ações (comandos), claras e precisas, com fluxo de execução baseado em:
 - sequência;
 - seleção condicional (seleção de ações);
 - iteração (repetição de ações);
 - possui dados de entrada;
 - produz dados de saída corretos;
 - possui estado final previsível;
 - deve ser eficaz.

Programação
Estruturada

ALGORITMOS



Não Numéricos

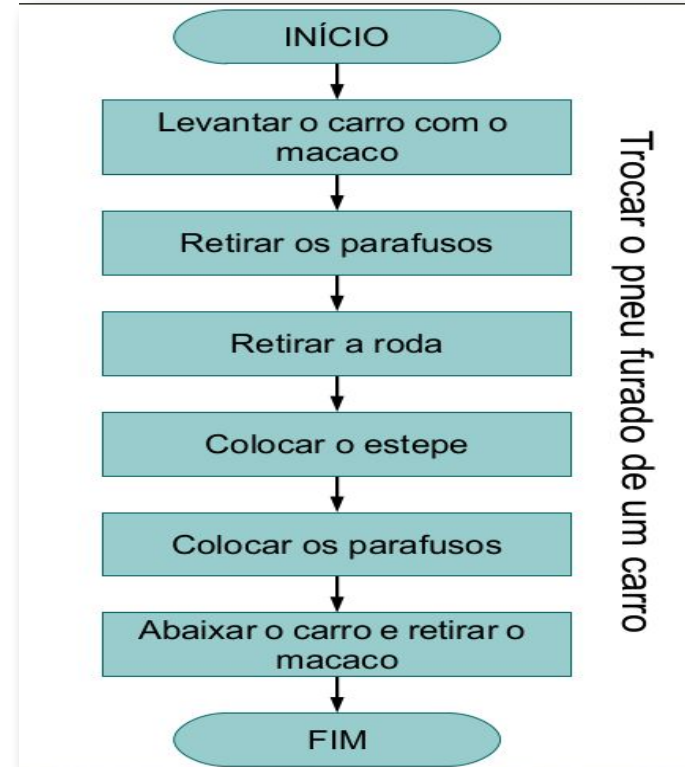
*Trocar o pneu furado
de um carro*

Numéricos

*Encontrar raízes de
uma equação*

ALGORITMOS — FLUXOGRAMA

- Diagrama de Blocos, Diagrama de Fluxo ou Fluxograma
 - Um método simples de retratar os passos de um procedimento numa sequência logicamente ordenada



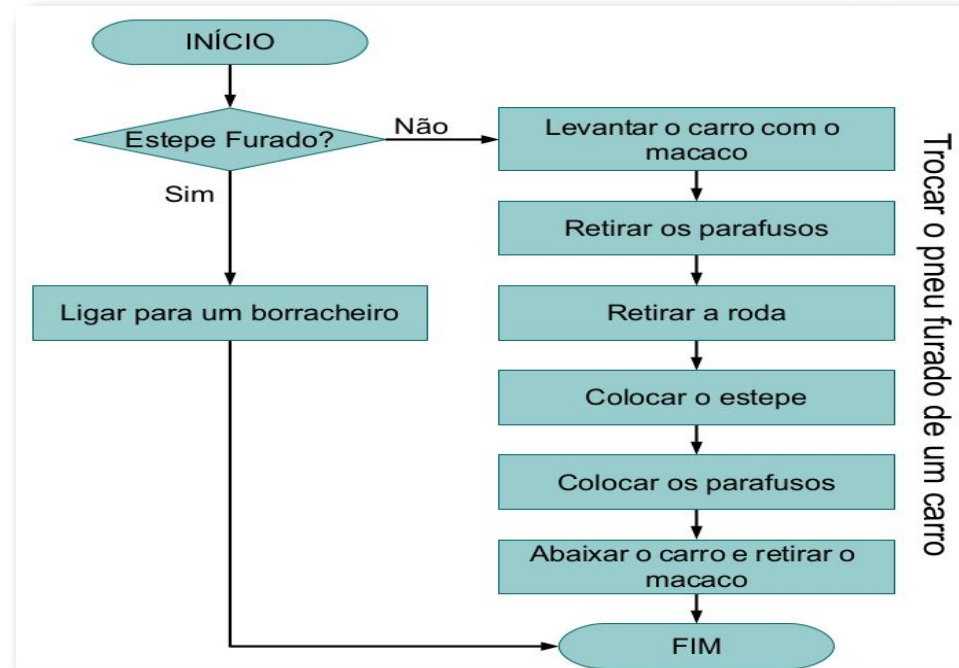
ALGORITMOS - FLUXOGRAMA

- E se houver necessidade de tomada de decisão?



ALGORITMOS – FLUXOGRAMA

- E se houver necessidade de tomada de decisão?



ALGORITMOS – FLUXOGRAMA



Fluxo, indica o sentido do fluxo de operações



Terminal, indica início, fim ou ponto de parada de um programa



Entrada/Saída, indica uma operação que utiliza uma unidade de entrada ou saída



Processamento, indica uma instrução ou grupo de instruções que realizam uma função de processamento



Decisão, indica a possibilidade de desvio para um ponto do programa, conforme condições estabelecidas



Conector, indica conexão entre 2 pontos do programa que estejam na mesma página



Conector, entre 2 pontos em páginas diferentes

ALGORITMOS NUMÉRICOS

- Seja a equação de 2o grau:
- As soluções são:
- Sendo:

$$ax^2 + bx + c = 0$$

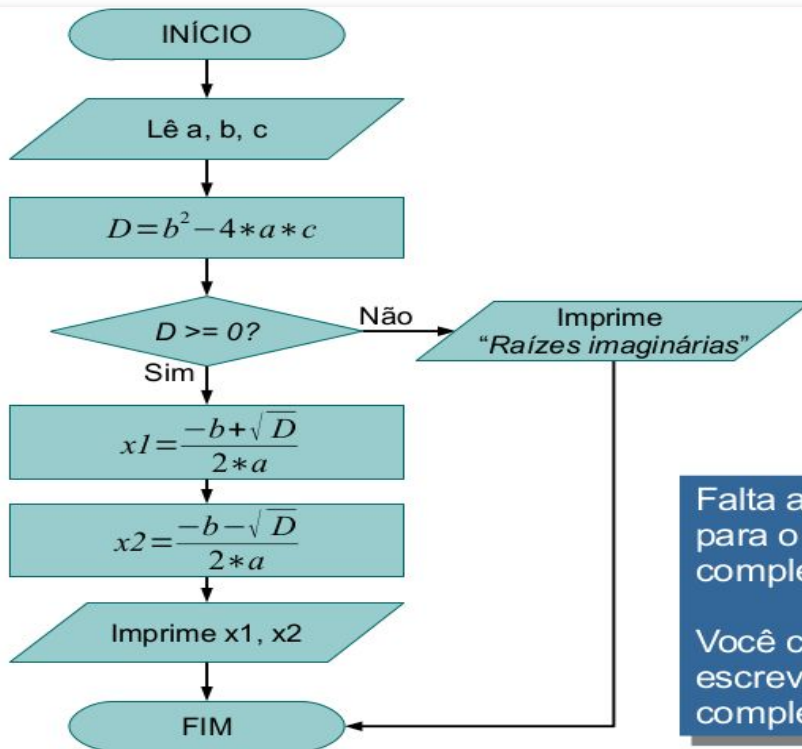
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

EXEMPLO DE ALGORITMO NUMÉRICO.

Caso D seja igual a zero, ele calcula x1, depois calcula x2 e imprime duas raízes iguais



Falta alguma condição para o algoritmo ser completo?

Você consegue escrever o fluxograma completo?

PSEUDOCÓDIGO

- Também conhecido como “Linguagem Estruturada” ou “Portugol”.
- Fortemente precisa, quase uma linguagem de programação real.

```
Algoritmo <nome_do_algoritmo>  
  <declaração_de_variáveis>  
  <subalgoritmos>  
  Início  
    <corpo do algoritmo>  
  Fim.
```

Considere o problema a seguir

Escreva um algoritmo para calcular a área de uma mesa retangular qualquer, considerando que os comprimentos da base e da altura são fornecidos pelo usuário

Resolvendo o Problema

1. Ler atentamente o enunciado, visando a sua compreensão da forma mais completa possível

Escreva um algoritmo para calcular a área de uma mesa retangular qualquer, considerando que os comprimentos da base e da altura são fornecidos pelo usuário



Resolvendo o Problema

2 . Analisar detalhadamente:

- a. que resultados devem ser obtidos?
- b. que dados são fornecidos?
- c. que procedimentos devem ser realizados para a
- d. solução do problema?

Resolvendo o Problema

2. Analisar detalhadamente:

- a. que resultados devem ser obtidos?
 - i. Cálculo da área da mesa
- b. que dados são fornecidos?
 - i. Valores para base e altura
- c. que procedimentos devem ser realizados para a solução do problema?
 - i. Área é igual a base x altura

Resolvendo o Problema

3. Construir o algoritmo
 - a) Estado Inicial
 - b) Estado Final
 - c) Dados de Entrada
 - d) Dados de Saída
 - e) Sequência lógica e finita de ações

Resolvendo o Problema

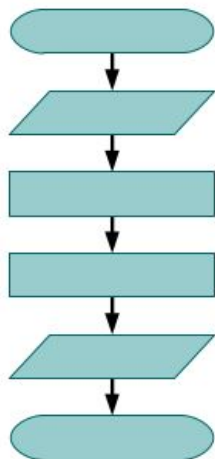
3. Construir o algoritmo
 - a) Estado Inicial
A área da mesa não é conhecida
 - b) Estado Final
A área da mesa é conhecida
 - c) Dados de Entrada
Base e Altura da mesa
 - d) Dados de Saída
Área da mesa
 - e) Sequência lógica e finita de ações
Fluxograma e Pseudocódigo

Resolvendo o Problema

4. Construir o algoritmo

Duas possibilidades de representação

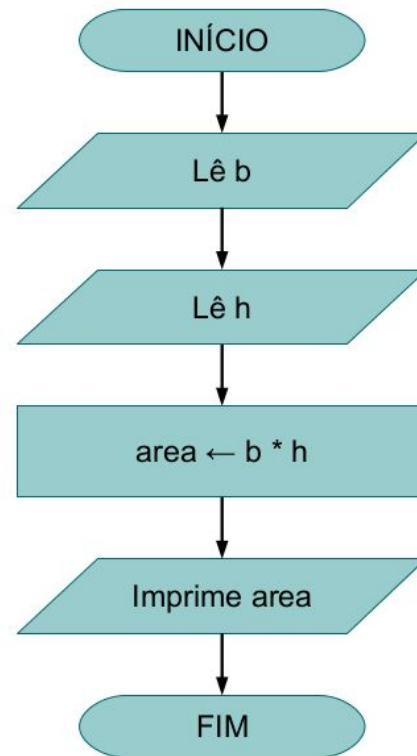
Fluxograma



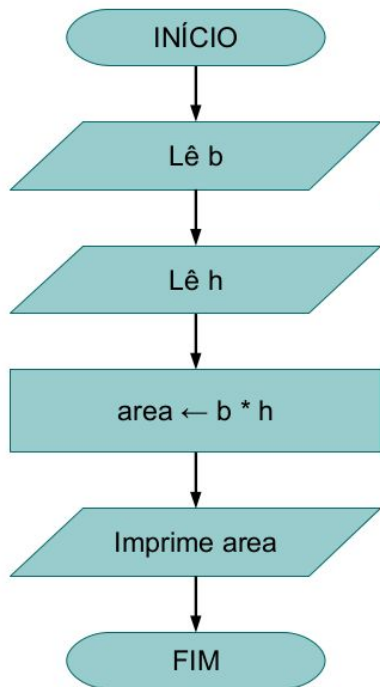
```
1 Pseudocódigo
2
3 Algoritmo
4
5 Início
6     inteiro a, b, b;
7     escreva("Digite o valor de a:")
8     leia(a)
9     escreva("Digite o valor de b:")
10    leia(b)
11    c = a + b;
12    escreva("O valor de C é: ", c)
13 FIM
```

Fluxograma

- Variáveis
 - **b** (base)
 - **h** (altura)
 - **area** (área)
- Operações de entrada e saída
 - **Lê** (Ler dados do usuário)
 - **Imprime** (Mostrar dados ao usuário)
- Operações aritméticas
 - ***** (multiplicação)
- Manipulação de variáveis
 - **←** (atribuição de valor)



Fluxograma



Algum problema em relação a entrada do usuário?

Quais valores o usuário pode informar?

Inicialmente assumimos usuários bem comportados, mas não vai ser sempre assim

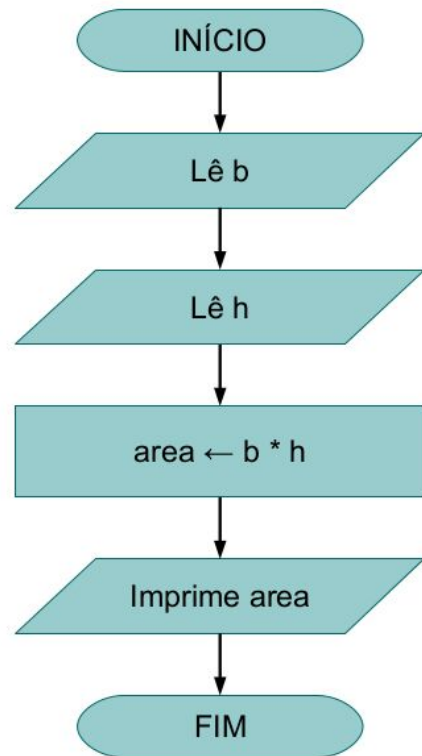
Pseudocódigo

- Variáveis
 - **b** (base)
 - **h** (altura)
 - **area** (área)
- Operações de entrada e saída
 - **Lê** (Ler dados do usuário)
 - **Imprime** (Mostrar dados ao usuário)
- Operações aritméticas
 - ***** (multiplicação)
- Manipulação de variáveis
 - \leftarrow (atribuição de valor)

```
1 Pseudocódigo
2
3 Algoritmo
4
5 Inicio
6     inteiro b, h, area;
7     escreva("Digite o valor da base:")
8     leia(b)
9     escreva("Digite o valor da altura:")
10    leia(h)
11    area = b * h;
12    escreva("A area é: ", area)
13 FIM
```

Pseudocódigo vs Fluxograma

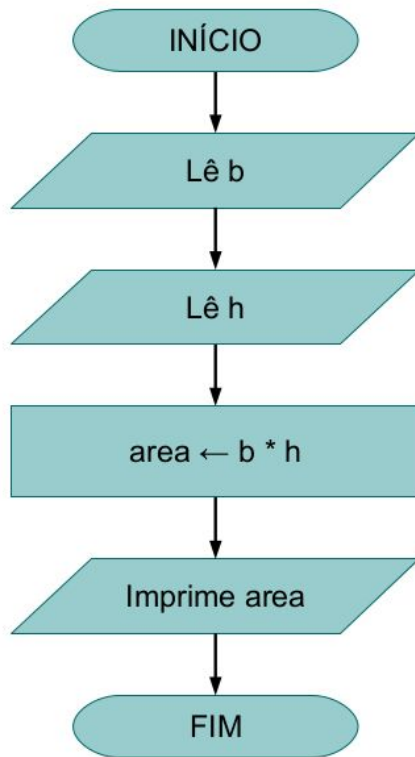
```
1 Pseudocódigo
2
3 Algoritmo
4
5 Inicio
6     inteiro b, h, area;
7     escreva("Digite o valor da base:")
8     leia(b)
9     escreva("Digite o valor da altura:")
10    leia(h)
11    area = b + h;
12    escreva("A area é: ", area)
13 FIM
```



Resolvendo o Problema (retomando)

4. Simular a sua execução ("**teste de mesa**")
 - Simular fluxo/sequência da execução
 - Controlar os valores das variáveis
 - No nosso exemplo: **b, h, area**
 - Gerar entradas do usuário
 - Analisar as saídas

Resolvendo o Problema (retomando)



Entradas

b ← 3

h ← 5

Variáveis

b

h

area

3

5

15

Saída

15

Proposta

Escreva um algoritmo para calcular a área e o perímetro de uma circunferência. O usuário deverá informar o raio apenas.

- Siga os passos descritos nessa aula para formular uma solução para o problema acima
- Dicas
 - Pi é uma constante aproximada (3.1415), não é uma variável
 - Área da circunferência
 - $a = \pi \cdot r^2$
- Perímetro da circunferência
 - $p = 2 \cdot \pi \cdot r$



Linguagens de programação: Conceitos, conceituação, classificação e paradigmas de linguagens de programação.

Conceito

- Uma LP é uma ferramenta utilizada para escrever programas.
- As primeiras linguagens de programação eram muito simples.
 - linguagem de máquina
- Com o avanço dos computadores e a necessidade de se desenvolver aplicações mais complexas, as LPs passaram a ser mais robustas.
 - C, C++, JAVA

Conceito

- Exemplo 1:
- Linguagem de alto
- nível

```
int A, B;  
int main()  
{  
    A = 2;  
    B = 1;  
    A = A + B;  
}
```

- Exemplo 2:
- Linguagem de baixo nível

```
MOV AX,0002  
MOV BX,0001  
ADD AX,BX
```


Conceito

- As linguagens de programação surgiram da necessidade de tornar o processo de desenvolvimento de software mais produtivo.
- Processo de desenvolvimento de software:
 - Especificação de requisitos;
 - Projeto do software;
 - Implementação;
 - Validação;
 - Manutenção.

Classificação

- Linguagem de máquina
 - Linguagem que a máquina é capaz de “entender” sem precisar ser traduzida.
 - Programas em linguagens de máquina são compostos por sequências de bits.
 - Ex.: 000110110
 - As sequências de bits referem-se a instruções e dados a serem executados.
-

Classificação

- Linguagem simbólica
 - Refere-se a uma abstração sobre as instruções e os dados.
 - As instruções são representadas por um “rótulo” simbólico que denota o verbo definido pela instrução.
 - Ex.: MOV AX, 0003
 - Um dado pode ser especificado diretamente na instrução ou pelo endereço de memória onde o dado é armazenado.
 - O endereço é denotado por um rótulo arbitrário definido pelo programador.
 - Programas em linguagem simbólica precisam ser traduzidos para outro equivalente em linguagem de máquina.
 - Os tradutores são chamados de montadores (assemblers).
 - Esse tipo de linguagem é utilizada em casos extremos onde é

Classificação

- Linguagem de alto nível
 - Em uma LP de alto nível o grau de abstração é bem alto, próximo ao modo de o ser humano pensar.
 - Nas LPs de alto nível os programadores trabalham com o conceito de comando e não de instruções de máquina, como acontece em LPs de baixo nível.
 - Um único comando em uma LP de alto nível pode se referir a vários comandos em uma LP de baixo nível.
 - O acesso à memória é realizado por meio de variáveis e constantes.
 - Os programas escritos em LPs de alto nível precisam ser traduzidos ou interpretados para serem executados.

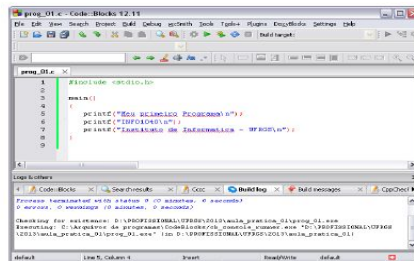
LINGUAGEM C

- Do código-fonte ao programa executável

```
#include <stdio.h>

main()
{
    printf("Meu primeiro Progr\n");
    printf("INFO1040\n");
    printf("Instituto de Inform\n");
}
```

Programa em
linguagem de programação
(código fonte)



Compilador
(ambiente de desenvolvimento)



Programa em
linguagem de máquina
(executável)

ESTRUTURA BÁSICA NA LINGUAGEM C

```
#include <stdio.h>
```

```
main()
```

```
{
```

```
}
```

Chamada às bibliotecas

→ *stdio.h* é usada para imprimir em tela

Função principal, *main()*

→ primeira função a ser executada
→ chaves delimitam o corpo da função

ADICIONANDO UM INSTRUÇÃO NA LINGUAGEM C

```
#include <stdio.h>

main()
{
    printf("Hello World.");
}
```

A função *printf* é usada para imprimir em tela.

```
Hello World.  
Process returned 12 (0xC)   execution time : 0.043 s  
Press any key to continue.
```

} Informações adicionais (tempo de execução, etc)

ADICIONANDO OUTRA INSTRUÇÃO NA LINGUAGEM C

```
#include <stdio.h>

main()
{
    printf("Hello World.");
    printf("inf 01040.");
}
```

```
Hello World.inf 01040
```


FORMATANDO A SAÍDA NA LINGUAGEM C

```
#include <stdio.h>

main()
{
    printf("Hello World.\n");
    printf("inf 01040.");
}
```

```
Hello World.
inf 01040
```

O “\n” efetua uma quebra de linha sempre que utilizado.

Deve estar entre aspas duplas do *printf*

FORMATANDO A SAÍDA NA LINGUAGEM C

```
#include <stdio.h>

main()
{
    printf("\nHello \nWorld.\n\n");
    printf("inf \n01040.");
}
```

```
Hello
World.

inf
01040
```

PSEUDOCÓDIGO X C

```
1 Pseudocódigo
2
3 Algoritmo
4
5 Inicio
6     inteiro a, b, b;
7     escreva("Digite o valor de a:")
8     leia(a)
9     escreva("Digite o valor de b:")
10    leia(b)
11    c = a + b;
12    escreva("O valor de C é: ", c)
13 FIM
```

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int a;
6     int b;
7     int c;
8     printf("Digite o valor de a: ");
9     scanf("%d", &a);
10    printf("Digite o valor de b: ");
11    scanf("%d", &b);
12    c = a + b;
13    printf("O valor de C é: %d", c);
14
15    return 0;
16 }
```

Ambiente de desenvolvimento

- https://www.tutorialspoint.com/compile_c_online.php - Compilador online
 - É necessário salvar o arquivo em no computador para conseguir executar em outro
 - https://www.onlinegdb.com/online_c_compiler
- Visual Studio + compilador C - tutorial:
 - <https://www.javatpoint.com/how-to-run-a-c-program-in-visual-studio-c-ode>