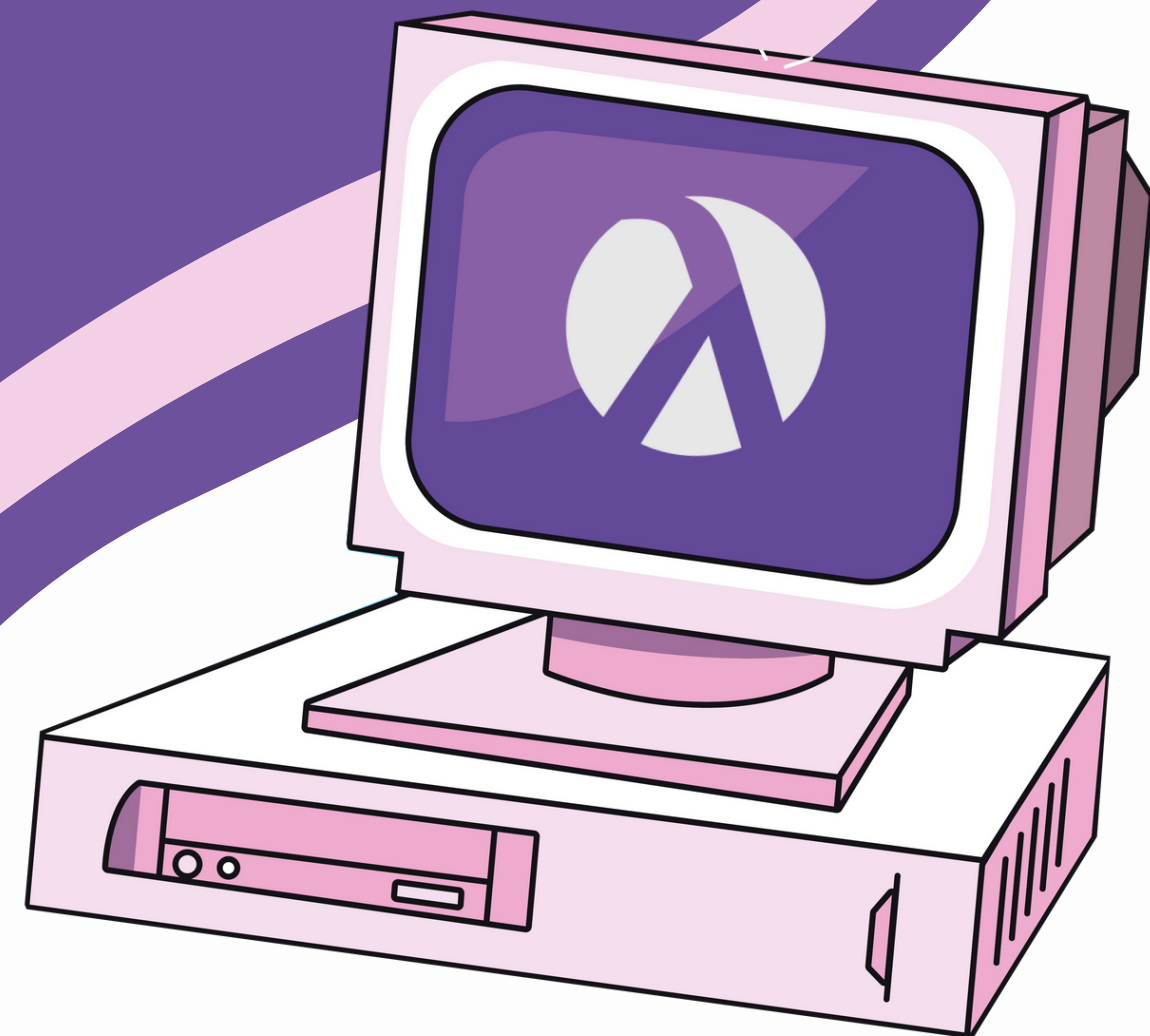


LÓGICA DE PROGRAMAÇÃO

EC

DIA 1



LÓGICA



Precisamos entender...

- **Lógica de programação**
- Algoritmo
- Linguagem



Lógica na programação



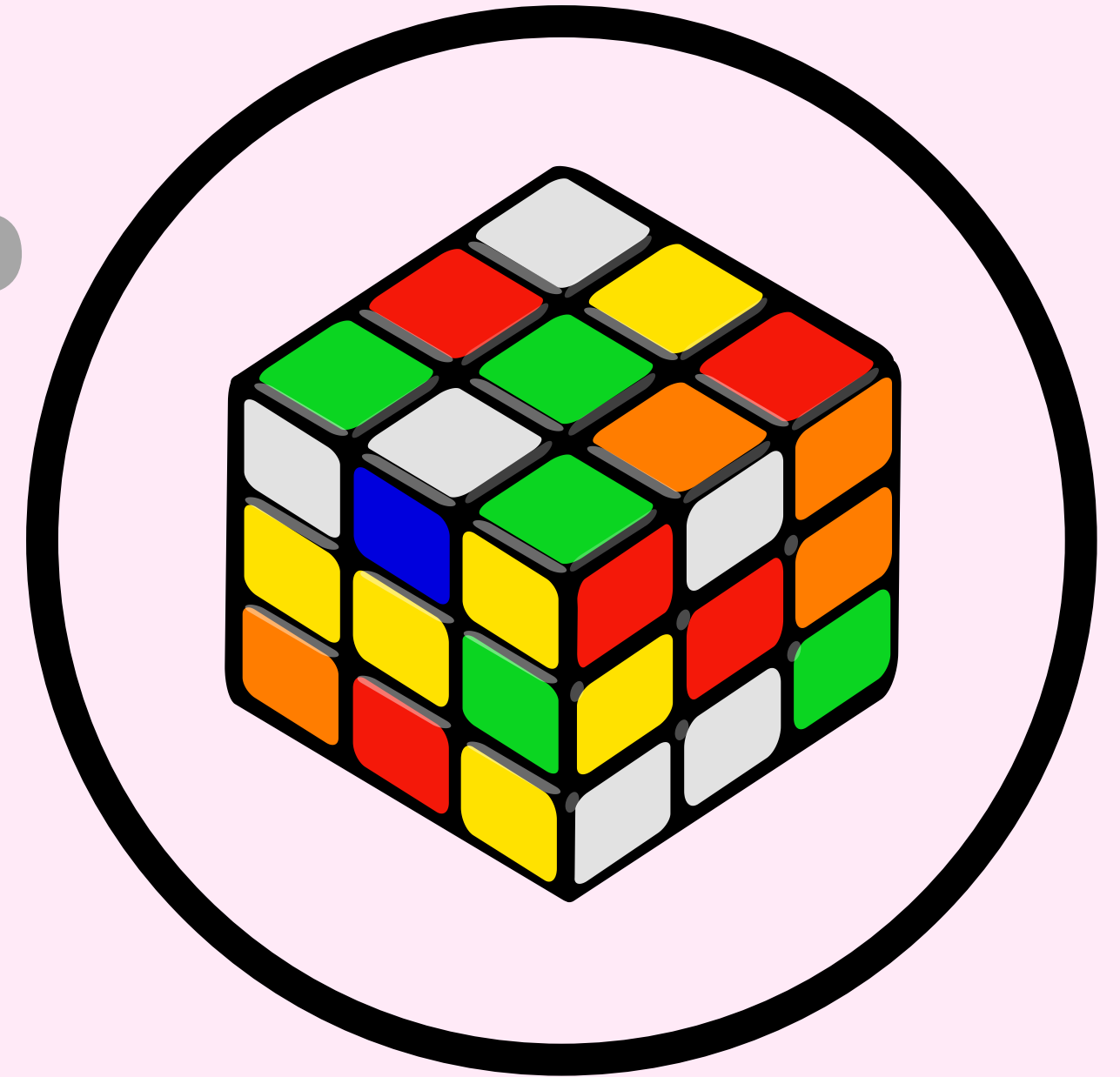
- Conceito geral para resolução de problemas
- Levando em conta a ordem dos passos
- Como o que vivemos no dia-a-dia



Precisamos entender...



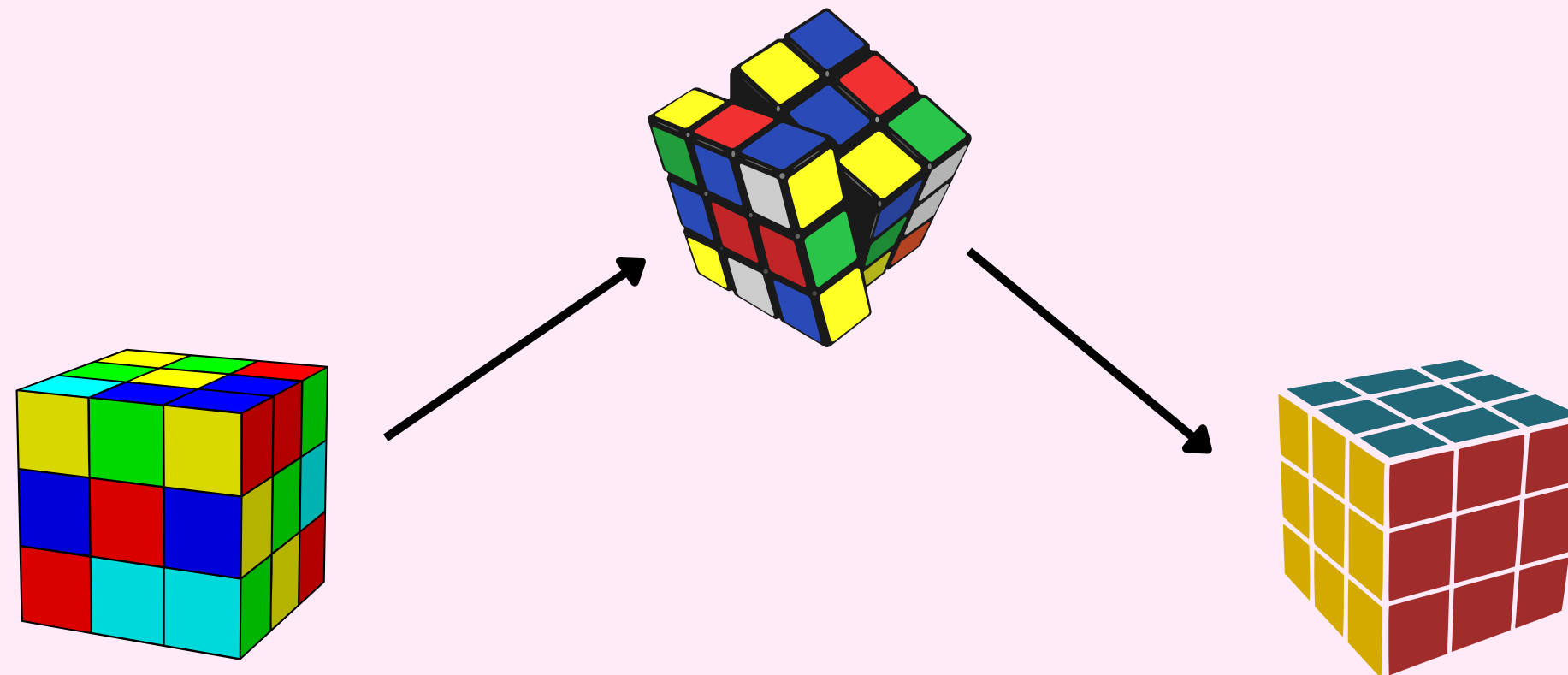
- Lógica de Programação
- **Algoritmo**
- Linguagem



Algoritmos



- Conjunto de regras e procedimentos lógicos que levam a uma solução de problema em um número finito de etapas



Algoritmos



- **Fazer omelete:**
 - **Misture os ovos**
 - **Quebre os ovos num pote**
 - **Desligue o fogão**
 - **Coloque os ovos misturados na frigideira**
 - **Ligue o fogão**
 - **Omelete pronta**

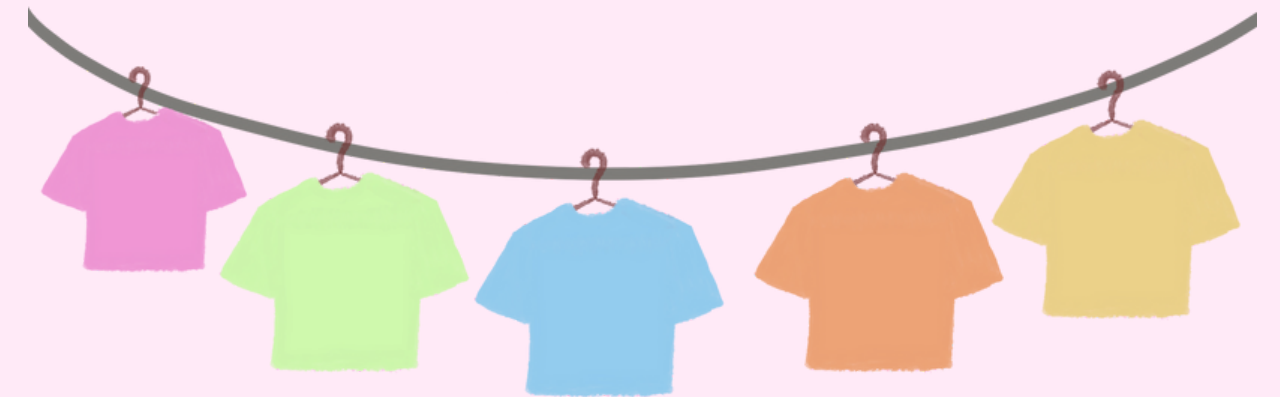
Agora funciona?



- **Fazer omelete:**
 - **Quebre os ovos num pote**
 - **Misture os ovos**
 - **Ligue o fogão**
 - **Coloque os ovos misturados na frigideira**
 - **Omelete pronta**
 - **Desligue o fogão**

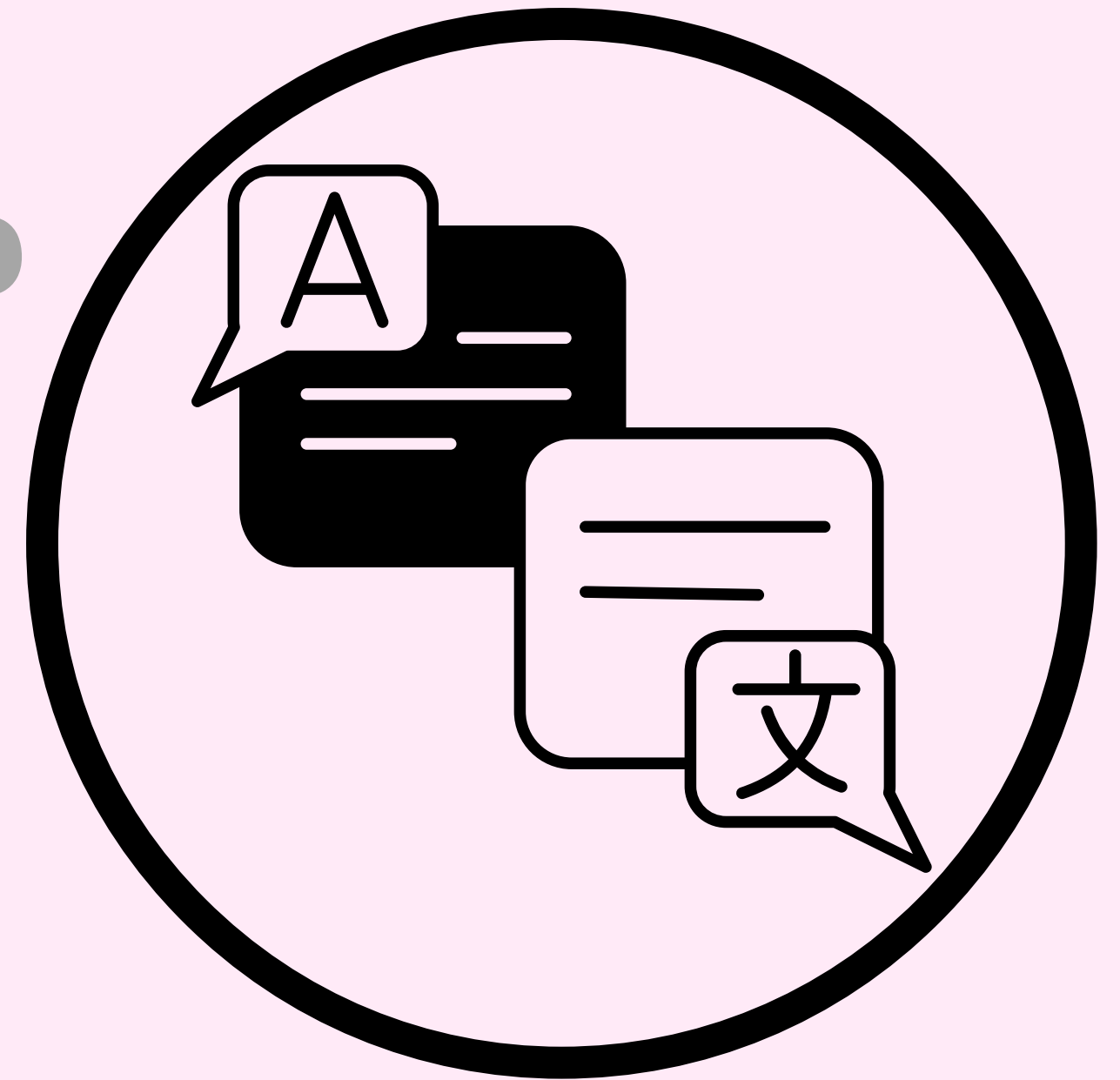
Agora você

- Lavar a roupa

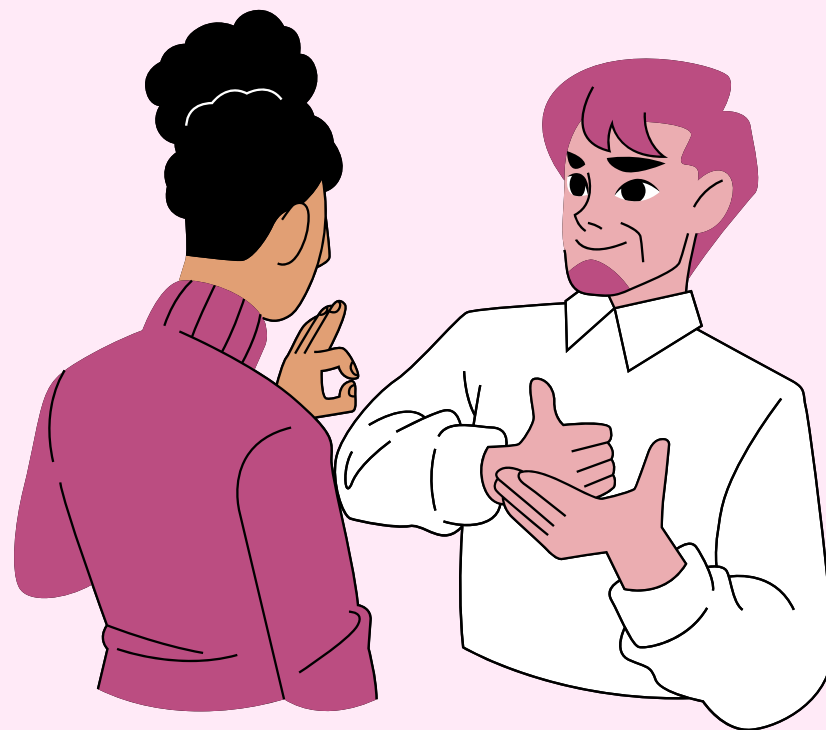


Precisamos entender...

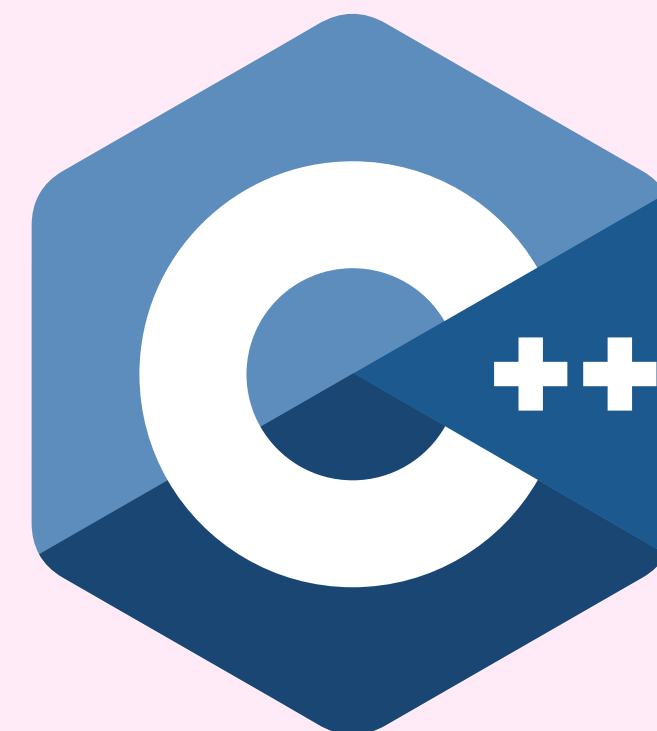
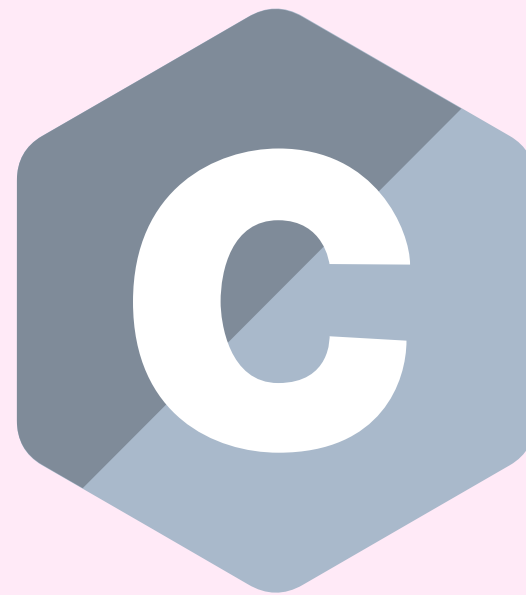
- Lógica de Programação
- Algoritmo
- **Linguagem**



Linguagem



Linguagem de programação



Linguagem C



- **Características**

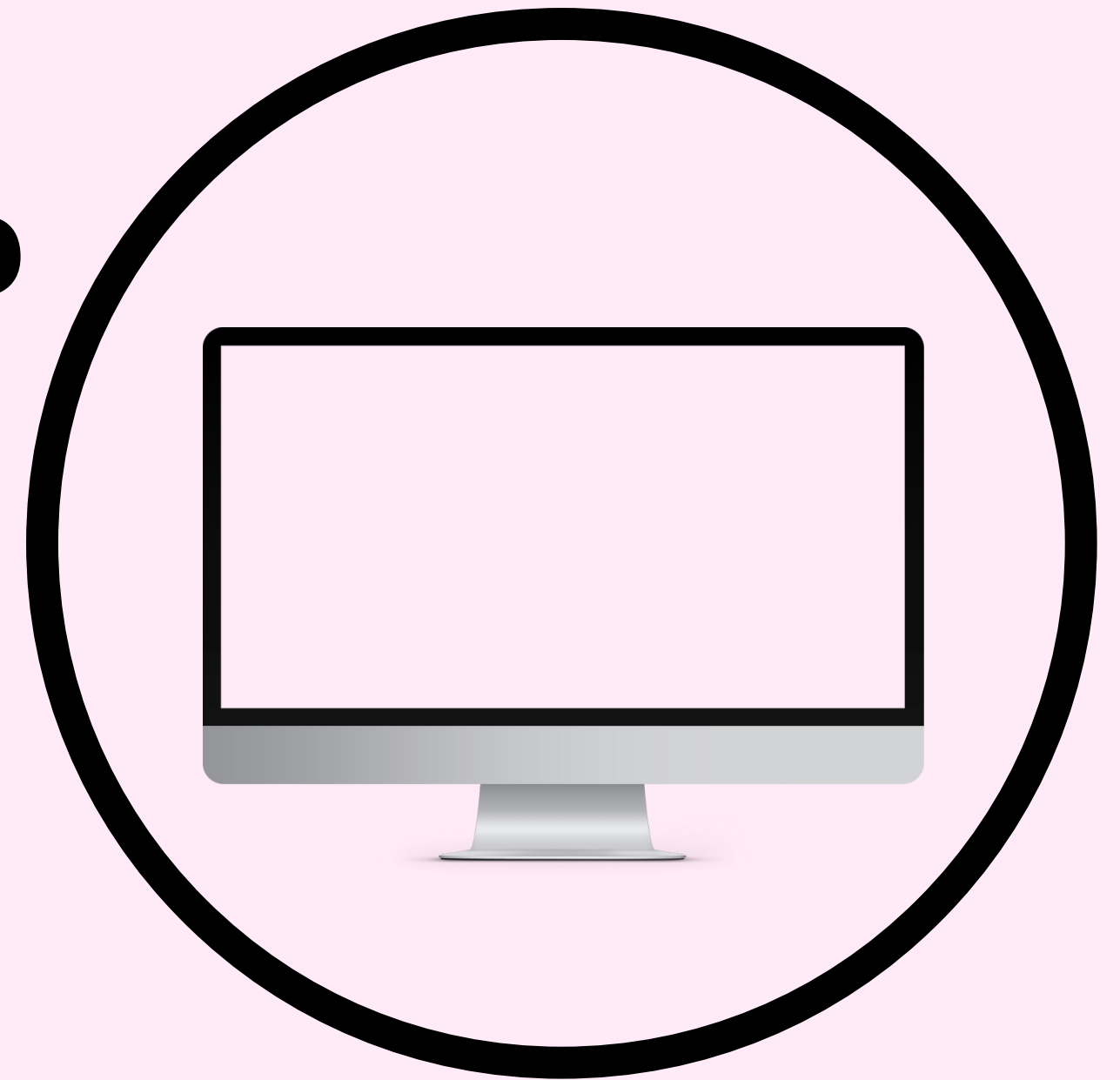
- Linguagem compilada de propósito genérico;
- Excelente performance;
- É uma linguagem que segue o paradigma de programação estruturada;
- Sintaxe de C serve de base para muitas outras linguagens.

PROGRAMAS DE COMPUTADOR



Já entendemos...

- **Lógica de programação**
- **Algoritmo**
- **Linguagem**



Programa de Computador

O que seria um programa?

- É a lógica do algoritmo implementada em uma linguagem de programação
- Ou seja, um conjunto de instruções de uma certa tarefa para ser realizada por um computador

```
1  #include <stdio.h>
2
3  int main() {
4      if (2 + 2 == 5) {
5          printf("Oi???");
6      } else {
7          printf("Ta tudo certo.");
8      }
9  }
10
```


Programa de Computador

Como o computador entende o programa?

- O computador trabalha em uma linguagem diferente do que a linguagem dos humanos, por isso a linguagem natural precisa ser traduzida para a Linguagem de Máquina.
- Compiladores e Interpretadores trabalham como intermediários entre o usuário e o computador, traduzindo a Linguagem de Programação para a Linguagem de Máquina.



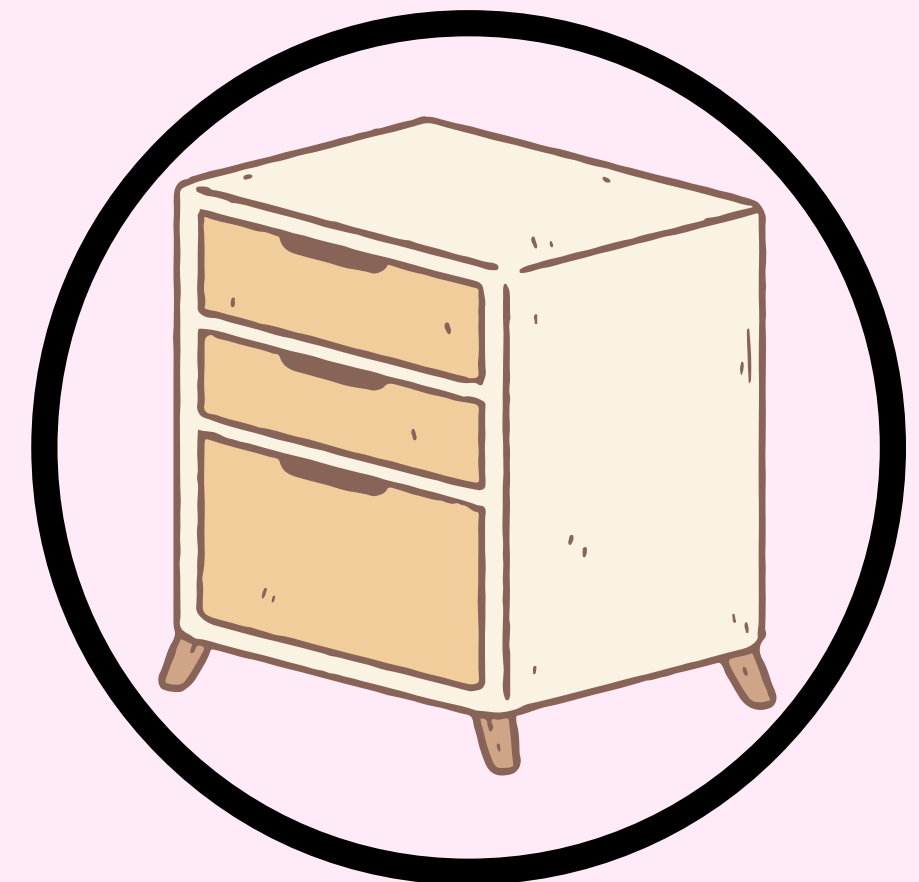
VARIÁVEIS E TIPOS



Variáveis

O que são:

- É um espaço reservado na memória RAM para guardar informações que serão utilizadas no programa
- Em uma variável, podemos alterar seu valor ao longo do programa, enquanto em constante o valor é "fixo"



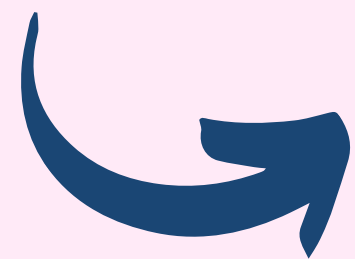
Variáveis

TODA VARIÁVEL **DEVE** SER DECLARADA ANTES DE SER USADA

```
int calouros;
```

```
int tutores;
```

```
int calouros, tutores;
```



Pode declarar várias
variáveis de um mesmo
tipo em uma só linha



Variáveis

Regras para nomear as variáveis em C:

- Não pode começar com número;
- Podem ser letras, números ou underline;
- A linguagem C é case-sensitive;
- Não podem ser uma palavra reservada.



Tipos das variáveis



TIPO	Descrição	Tamanho
char	Um caractere (ASCII)	1 byte
int	Inteiro	4 bytes
float	Ponto flutuante	4 bytes
long	Inteiro	8 bytes
double	Ponto flutuante	8 bytes

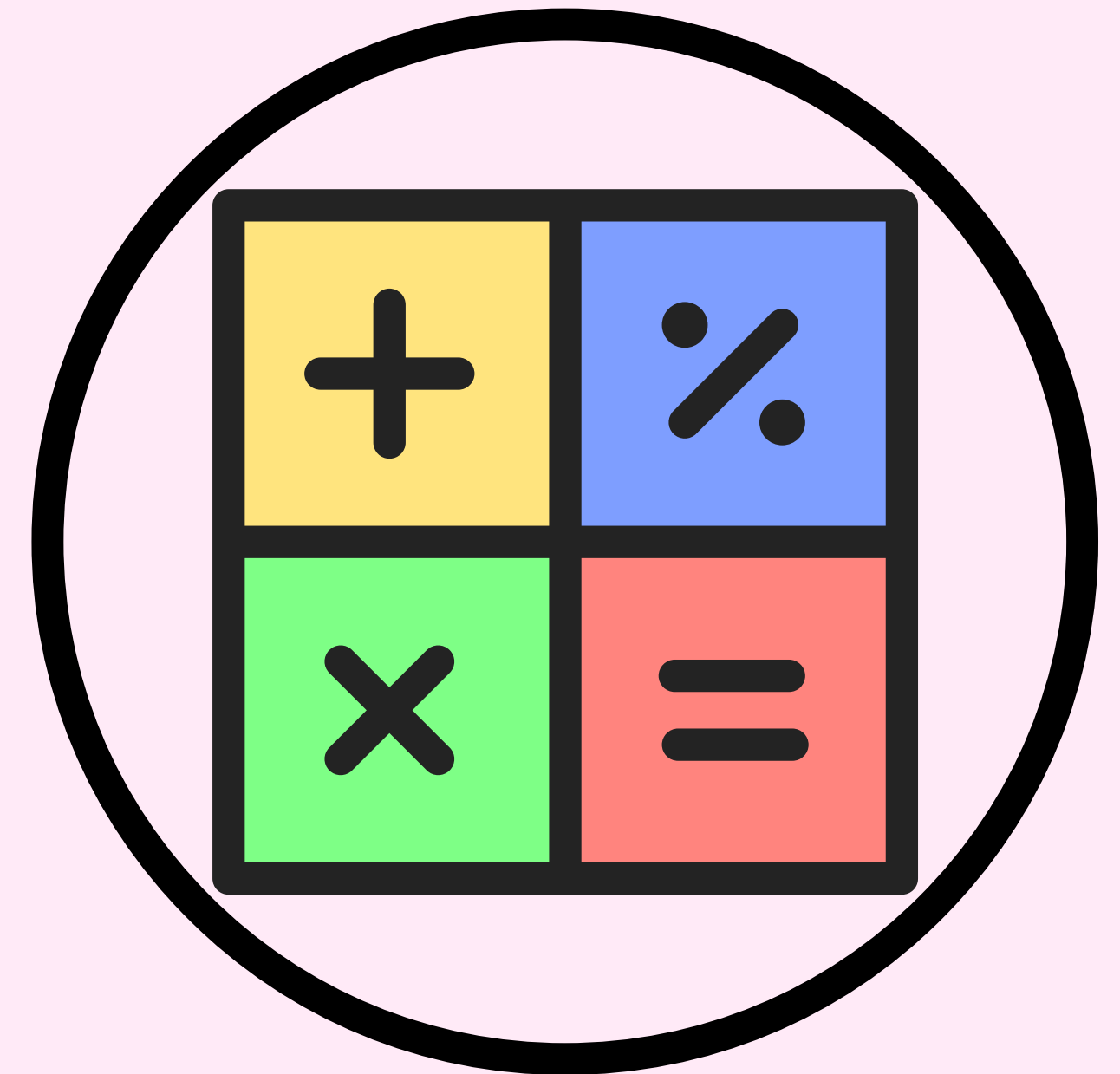
OPERADORES



OPERADORES

São símbolos associados a determinadas operações:

- **Operador de Atribuição;**
- **Operadores Aritméticos;**
- **Operadores Relacionais;**
- **Operadores Lógicos;**
- **Operadores de Incremento e Decremento.**



Operadores Aritméticos



+	Soma
-	Subtração
*	Multiplicação
/	Divisão
%	Resto da Divisão

```
int var1 = 10;
```

```
int var2 = 12;
```

```
int var3 = 8;
```

```
int var4 = 12 + 10;
```

```
var3 = var1 - var2;
```

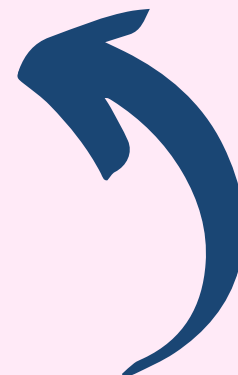
Operador de Atribuição



- Atribui um valor a uma variável;
- Sempre ocorre da direita para a esquerda.

Esse é o operador
de atribuição

int var1 = 10;

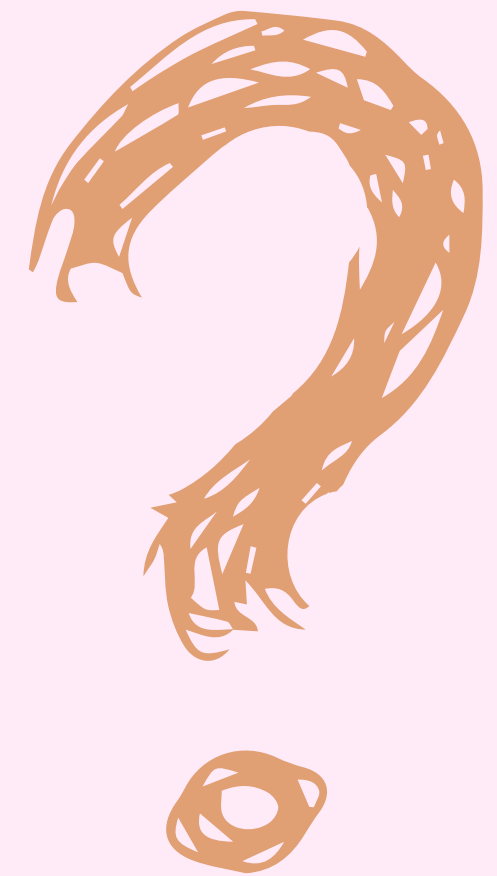


PERGUNTA

```
int var1 = 10;  
int var2 = 12;  
int var3;
```

```
var3 = var2;  
var1 = var3;
```

Qual o valor de var1?



Operadores Relacionais



==	Igualdade
!=	Diferente
<	Menor
>	Maior
<=	Menor ou Igual
>=	Maior ou Igual

- **Operadores relacionais são utilizados para comparar valores;**
- **O resultado de uma expressão relacional é um valor booleano (Verdadeiro ou Falso).**

Operadores Lógicos

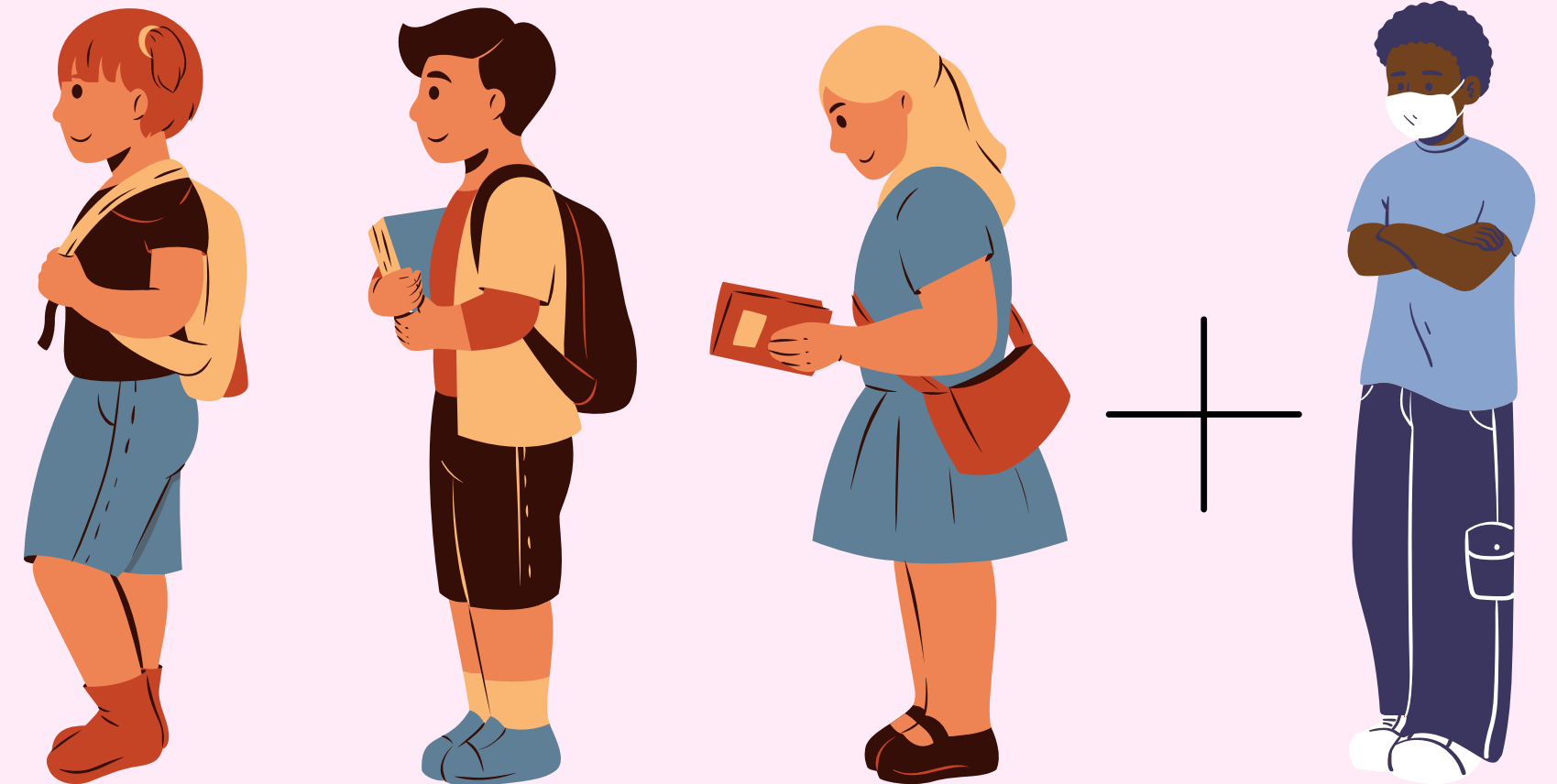


!	"NEGAÇÃO"
&&	"E"
 	"OU"

Operadores de Incremento e Decremento



++	Incremento
--	Decremento



ENTRADAS E SAÍDAS



Estrutura Básica do Programa



Função de
Início de
Programa.

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

```
int main () {  
    /* ... */  
    return 0;  
}
```

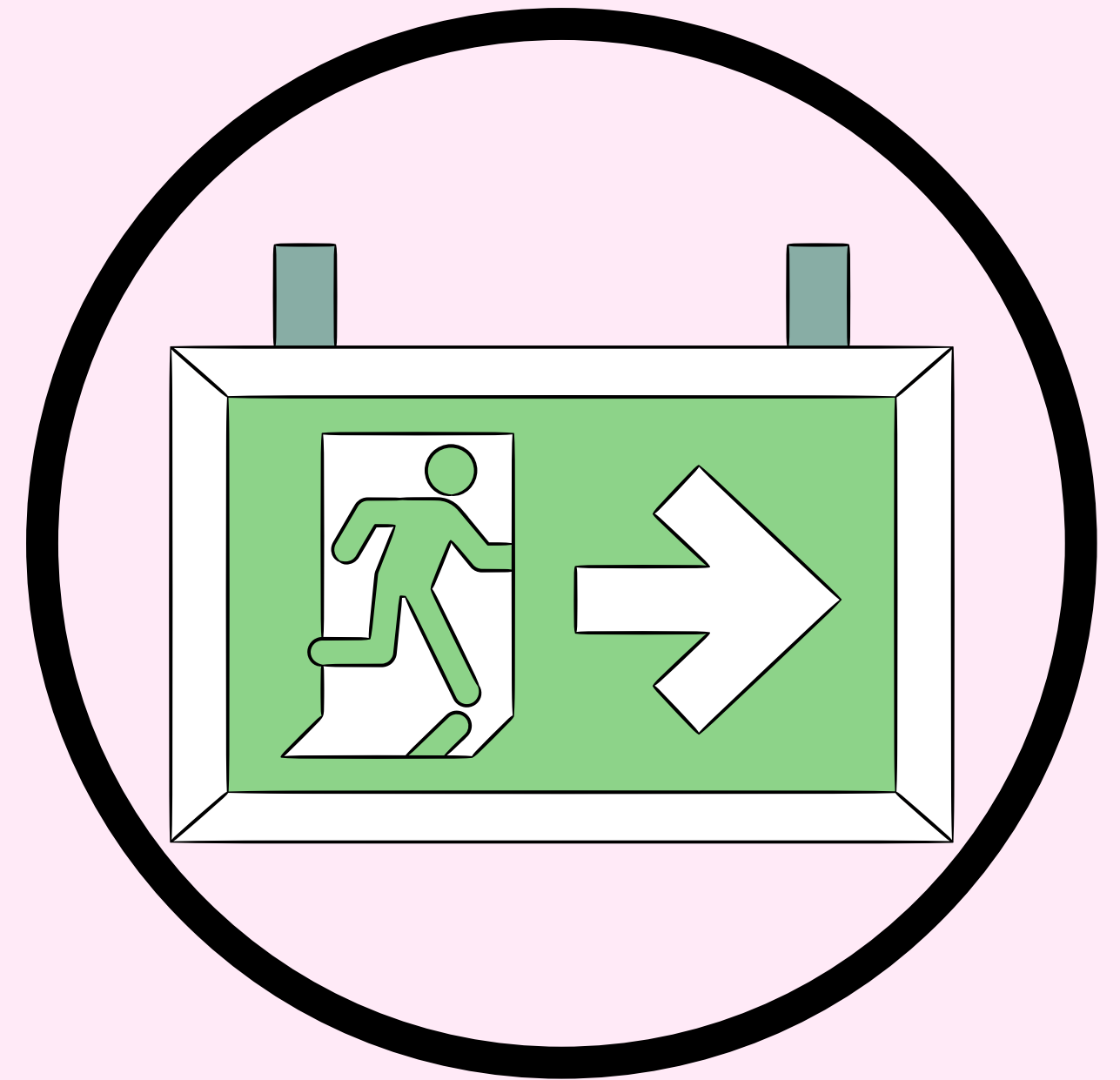
Declarações globais e
bibliotecas usadas pelo
programa.

Comentário indicando o local
onde outras instruções devem
ficar.

Instrução de retorno seguido pelo
valor a ser retornado.

Funções de saída

- **printf**
- **putchar**
- **puts**



PRINTF

Mostra na tela o valor de qualquer tipo de dado

```
int qtdPessoas = 20;
```

```
float nota = 9.75;
```

```
char classificacão = 'L';
```

```
char nome[20] = "Shrek";
```

```
printf("%d pessoas deram nota maior que %.2f  
para o filme %s, cuja classificacão indicativa é  
%c", qtdPessoas, nota, nome, classificacao);
```



Entrada/Saída



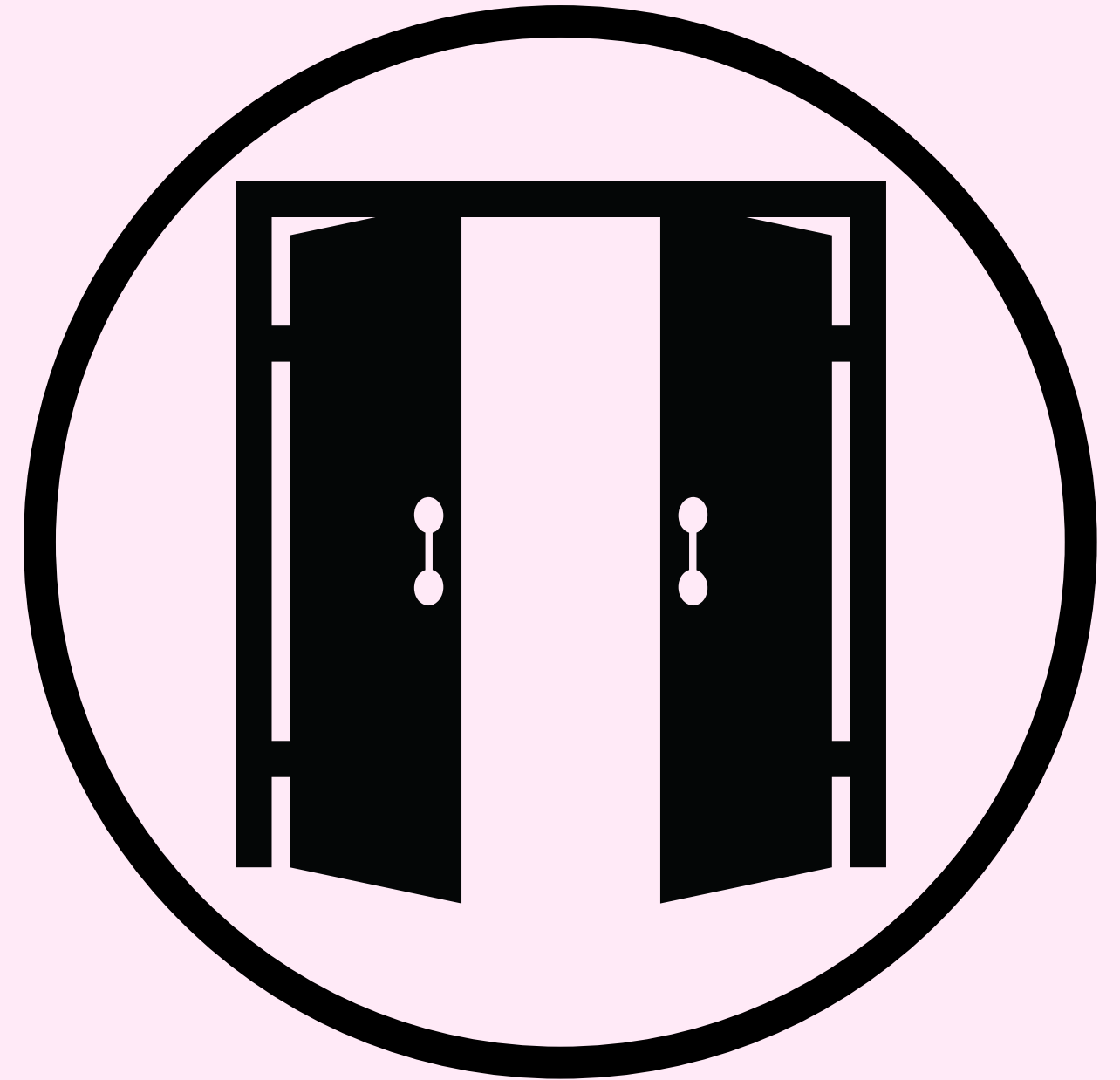
Tabela dos formatos

TIPO	Descrição	Formato
char	Um caractere (ASCII)	%c
string	cadeia de caracteres	%s
int	Um inteiro	%d ou %i
float	Ponto flutuante	%f

Funções de entrada



- **scanf**
- **getchar**
- **gets**



SCANF

LÊ VALORES DO USUÁRIO

```
int idade;  
scanf("%d", &idade);
```

```
float altura;  
char tamanhoCamiseta;
```

```
scanf("%f %c", &altura, &tamanhoCamiseta);
```

```
char nome[20];  
scanf("%s", nome);
```

string
não precisa de &



Hora de botar a mão na massa!



Crie uma interface de entrada e saída (scanf e printf), na qual o usuário deve fornecer os seguintes dados:

- **Idade**
- **Ano de nascimento**
- **RA**

O programa deve mostrar os dados fornecidos na tela.

Resposta



```
1  #include <stdio.h>
2
3  void main()
4  {
5      int idade, anoNasc, ra;
6
7      printf("Qual a sua Idade?\n");
8      scanf("%d", &idade);
9
10     printf("\nEm que ano voce nasceu?\n");
11     scanf("%d", &anoNasc);
12
13     printf("\nQual o seu RA?\n");
14     scanf("%d", &ra);
15
16     printf("\nIdade: %d, Ano de Nascimento: %d, RA: %d", idade, anoNasc, ra);
17 }
```

Hora de botar a mão na massa!



Crie um algoritmo para calcular a média de um aluno

O usuário deve fornecer os seguintes dados:

- **nota1, nota2 e nota3**
- **nome**

O programa deve mostrar na tela o nome e a média desse aluno

Resposta



```
1  #include <stdio.h>
2
3  int main() {
4      char nome[20];
5      float nota1, nota2, nota3;
6
7      printf("Digite o nome do aluno: ");
8      scanf("%s", nome);
9
10     printf("Digite a nota de %s na Prova 1: ", nome);
11     scanf("%f", &nota1);
12
13     printf("Digite a nota de %s na Prova 2: ", nome);
14     scanf("%f", &nota2);
15
16     printf("Digite a nota de %s na Prova 3: ", nome);
17     scanf("%f", &nota3);
18
19     float media = (nota1 + nota2 + nota3) / 3.0;
20     printf("A media do aluno %s foi: %.2f\n", nome, media);
21 }
22
```

PROJETINHO



Projeto

- Criar variáveis com os nomes dos produtos que estarão disponíveis para compra e o valor deles dentro do sistema do mercado.

```
float PrecoPacoteDeArroz = 12.5;  
float PrecoPacoteDeFeijao = 8;  
float PrecoCaixaDeOvos = 6.5;  
float PrecoRefrigerante = 5.5;  
float PrecoSalgadinho = 3.5;  
  
float valorCompra, troco, valorRecebido;  
  
int QntPacoteDeArroz, QntPacoteDeFeijao, QntCaixaDeOvos, QntRefrigerante, QntSalgadinho;
```

Projeto



- Criar uma interface de entrada e saída de informações para que o caixa possa digitar a quantidade de produtos que o cliente irá comprar.
- Mostrar no final o valor total da compra feita pelo cliente.
- Criar uma entrada para o valor recebido e exibir o troco para ser devolvido para o cliente.