

A person wearing a dark suit and black shoes is walking from left to right. The background is a chalkboard filled with mathematical content. On the left, there's a coordinate system with a vertical axis labeled 'b' and a horizontal axis. In the center, there are several equations: $(\frac{1}{\sin \pi})^2$, $-\infty + \infty$, and $(\ln x - \cos y)^2 + 2$. On the right, there are two 3D wireframe diagrams of rectangular prisms. The first prism has a diagonal line drawn from the bottom-left corner to the top-right corner, labeled with the expression $y^2 + 18^2 = 1^2$. The second prism is similar but has a different internal line drawn. The text "Primeiros Passos" is overlaid in red at the bottom right.

Primeiros Passos

Conferir a data e o horário do filme no bilhete.

Não permitir a entrada.

Não permitir a entrada.

Informar ao cliente que o tempo limite para entrada foi excedido.

Senão Se data/hora atual < data/hora do filme - 30 minutos Então

Indicar ao cliente onde fica a sala do filme.

Se data/hora atual > data/hora do filme + 30 minutos Então

Informar ao cliente que a sala do filme ainda não foi liberada para entrada.

Solicitar ao cliente o bilhete do filme.

Senão

Permitir a entrada.

Se estacionou, retire a chave da ignição e saia do carro.

Se a garagem estiver aberta, saia. Senão, abra-a e saia.

Entre no carro.

Se a lanchonete estiver aberta, entre e dirija-se ao caixa. Senão, volte para o carro. Procure outra lanchonete aberta, entre e dirija-se ao caixa.

Faça o pedido e efetue o pagamento. Aguarde.

Se chegou, estacione o carro. Senão, continue até chegar e estacionar.

Se o número de seu pedido for chamado, dirija-se ao caixa e pegue sua refeição. Senão, aguarde ser chamado e pegue-a.

Tranque o carro e dirija-se à lanchonete.

Saia de casa.

Programação

Processo de desenvolvimento de atividades ou instruções

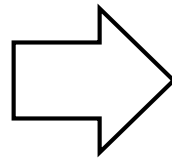
3 Etapas:

- Análise - Reflexão sobre o problema → entradas, saídas, processo
- Algoritmo - Definição dos passos do processo de resolução
- Codificação - Transformação do algoritmo em códigos

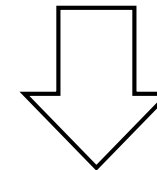
O compilador

Ponto de Origem

Código fonte
descrito numa
linguagem de alto-
nível (ex.: C, C++,
Pascal, etc.)



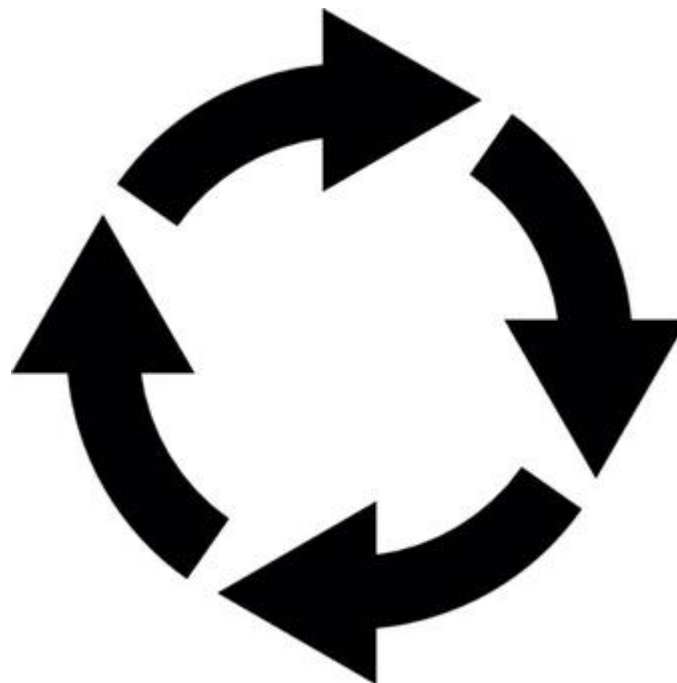
Compilador



Programa em
Linguagem Máquina

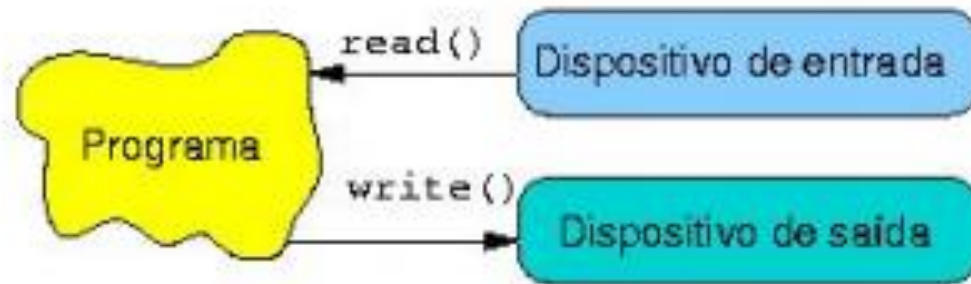
Ponto de Destino

A comunicação



Entrada X Saída

- Entradas
 - Dados necessários para a resolução do problema
- Saída
 - Dados computados pelo programa ou mensagens a serem exibidas



Entrada X Saída

	Início do Algoritmo
	Leitura de um valor do teclado
	Atribuição de valores
	Escrita de um valor na tela
	Final de um algoritmo

Como representar as informações?

Dados

- Informações a serem processadas e/ou armazenadas
- Somente binários?

3 tipos primitivos

- **Numéricos** → inteiros (-352; 0; 5) ou reais (10.55; -152.7; 0.0)
- **Caracteres ou literais** (“A maçã está estragada”, “15678”, “A”, “#@!”)
- **Lógicos** (Verdadeiro ou Falso; 0 ou 1)

VARIÁVEIS

Valor que pode ser alterado

- Precisa de uma identificação e de um tipo para o dado vinculado

- Identificador (nome, etiqueta...)

- Deve ser único
- Não pode ter espaços em branco
- Não pode ter símbolos especiais exceto o underscore: _
- Deve iniciar com uma letra
- Não pode ser uma palavra reservada da linguagem de programação
- Recomendável: usar um nome que lembre do que se trata a variável

- Exemplos

- Salário
- 1ano
- ano3
- t3t0
- aluno
- NoTa
- x,y,z
- raiz_quadrada

VARIÁVEIS-CONSTANTES

Constante

- Valor fixo que não pode ser alterado

- Exemplos

$\pi = 3.14159\dots$

velocidade da luz = $299792.458 \text{ km / s}$

Usando uma variável

- Declaração

```
TIPO nome_variavel;
```

```
TIPO v1,v2,v3;
```

```
TIPO v4=VALOR;
```

- A declaração de variáveis pode vir a qualquer momento
 - Desde que seja antes da utilização

```
int x;
```

```
x = 8;
```

```
int y;
```

```
y = x+6;
```

Alguns tipos em C

- `int` → número inteiro
 - 4 bytes
 - -2,147,483,648 até 2,147,483,647
- `float` → número real
 - 4 bytes
 - precisão de até 6 casas decimais
- `char` → caracter
 - 1 byte
 - 0 até 255
- `bool` → valor lógico
- `void`
 - tipo nulo específico para funções
 - indica que a função não retorna valor

Exemplos

```
int x = 8;  
float total = 105.56;  
char letra = 's';  
bool resp = true;  
#define pi 3.141592  
int x, y, z=0;  
x=1;  
y=z=x;  
x=z+1;  
y=z-x;
```

A memória

Representação do Endereçamento	
Memória	Variável
001	x
002	y
003	nome
004	média

Representação da Memória e os Valores	
Memória	Valor
001	30
002	15.9
003	"UFC"
004	8.0

Código C

```
#include<stdio.h>

int main(void){
    int x;
    float y, media;
    char nome[10] = "UFC";

    x = 30;
    y = 15.9;
    media = 8.0;
}
```

Modificando as variáveis por operações

Expressões Aritméticas

- Expressões formadas por números, constantes e variáveis através de operadores aritméticos que resultam em um valor numérico
- Operadores aritméticos básicos: + - * /

- Exemplos

- $2 + 3 * a - (b / 4)$
- $((b + 4.2) * x) / (2 * \text{pi})$

Modificando as variáveis por operações

Expressões Lógicas

- Expressões formadas através de operadores lógicos que resultam em um valor lógico (verdadeiro ou falso)
- Operadores lógicos básicos: = < > !
- Exemplos
 - $43 < 2.5$
 - $x > 75$

Regras!

A diretiva `#include` carrega uma biblioteca para ser usada

Sensível a letras maiúsculas e minúsculas

- `include` \neq `Include` \neq `INCLUDE`

Todo comando deve terminar com ponto-e-vírgula (;)

```
#include <stdio.h>

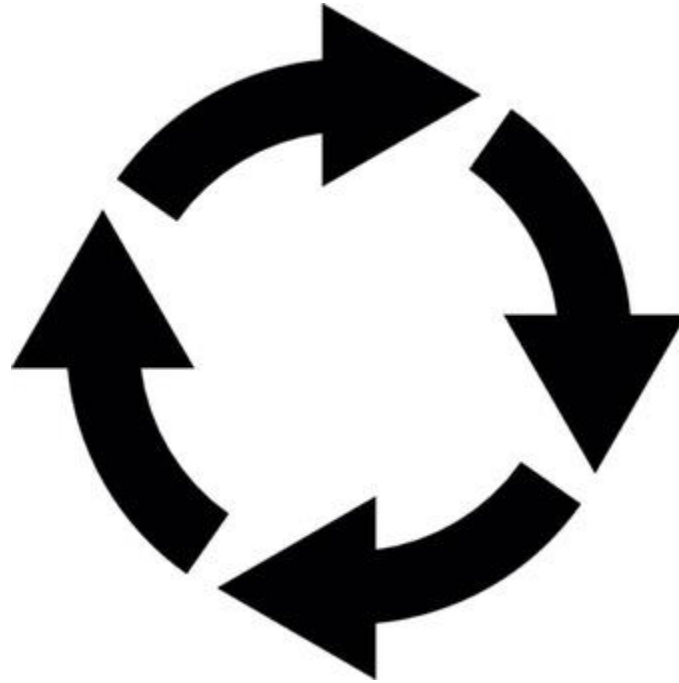
void main( )
{
    printf("Hello");
}
```

scanf e printf



scanf → função para entrada de dados pelo teclado

- `scanf(codigo, &variavel);`



printf → função para saída de dados para o console

- `printf(string, &valor);`

Um teste

Código	Formato
%c	Caractere
%i	Inteiro
%l	Inteiro longo
%d	Inteiro em formato decimal
%ld	Decimal longo
%u	Decimal sem sinal
%f	Ponto flutuante em formato decimal
%lf	Ponto flutuante longo (<i>double</i>)
%s	Cadeia de caracteres (<i>string</i>)
%e	Valor em notação científica
%o	Valor em notação octaldecimal
%x	Valor em notação hexadecimal
%p	Ponteiro
%%	Símbolo %

A prática

1. Faça um programa que receba dois números e ao final mostre a soma, subtração, multiplicação e a divisão dos números lidos.
2. Escreva um programa dobro do mesmo.
3. Faça um programa que leia dois números e troque o conteúdo entre as variáveis que armazenam os valores
4. Faça um programa que leia uma temperatura em graus Fahrenheit e converta-a para graus centígrados através da fórmula $C = 5/9 (F - 32)$.

5. Faça um programa para solicitar os lados de um retângulo, determinar e mostrar sua área e seu perímetro.
6. Faça um programa que leia valores para duas variáveis, exibindo-as e, em seguida, troque os valores das mesmas, exibindo-as novamente. O resultado deve ser exibido assim:
 1. Valores lidos.....: $A = 5$ e $B = 12$
 2. Valores trocados: $A = 12$ e $B = 5$
7. Escreva um programa para ler um comprimento dado em centímetros e exibir quantos metros.