

Curso de Programação em Computadores V

Módulo 1 -Algoritmos e Programação

Aura Conci & Erick Oliveira
aconci@ic.uff.br, erickr@id.uff.br

Disciplina: Programação V

▶ Bibliografia básica:

- ▶ W. Celes, R. Cerqueira e J. L. Rangel, **Introdução a estruturas de dados, com técnicas de programação em C** , Ed. Campus, 2004
- ▶ B. W. Kernighan e D. M. Ritchie , **C a linguagem de Programação**, Ed. Campus, 1978
- ▶ T. Swan, **Tece e aprenda C**, Berkley, 1994
- ▶ H. Schildt, **C avançado – guia do usuário**, Mc Graw-Hill, 1989.
- ▶ Slides dos modulas(aulas)



Objetivos da Disciplina

- ▶ Solucionar problemas de forma virtual
- ▶ Programar em C
 - ▶ Fácil partir para outras linguagens se necessário



Forma de Avaliação

- ▶ 2 provas (P1 e P2) + médias dos trabalhos (T)
- ▶ Média = $(P1 + P2 + T)/3$
- ▶ Critério para aprovação:
 - ▶ Média > 6



Roteiro da Aula

- ▶ Resolução de Problemas
- ▶ Algoritmo
- ▶ Modelo de um computador
- ▶ Elementos do algoritmo
 - ▶ Constantes e Variáveis
 - ▶ Atribuição
 - ▶ Operadores
- ▶ Ciclo de Vida de um programa
- ▶ Meu primeiro programa em C

Problema

- ▶ Fazer um omelete
- ▶ Qual a sequência de passos necessária para atingir o objetivo?



Passos para Fazer um Omelete

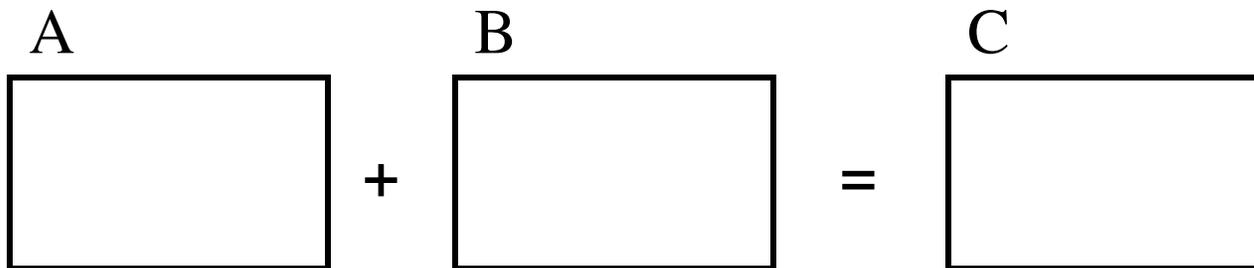
1. Quebrar os ovos
2. Bater os ovos
3. Adicionar sal
4. Ligar fogão
5. Adicionar óleo na frigideira
6. Colocar frigideira no fogo
7. Por os ovos batidos na frigideira
8. Verificar se está pronto
9. Se sim, desligar o fogão
10. Se não, voltar para passo 8

Observações Importantes

- ▶ Quanto às instruções isoladas:
 - ▶ Só “quebrar ovos”, ou só “pôr óleo na frigideira”, não é suficiente para cumprir a tarefa “fazer omelete”
- ▶ Quanto à seqüência lógica:
 - ▶ Se executarmos “por os ovos batidos” antes de “bater ovos”, ou pior, antes de “quebrar ovos”, não iremos cumprir a tarefa “fazer omelete”

Algoritmo

- ▶ Seqüência finita de passos que levam à execução de uma tarefa
- ▶ Claro e preciso. Ex. “somar dois números”:
 - ▶ Escrever primeiro número no retângulo A
 - ▶ Escrever segundo número no retângulo B
 - ▶ Somar o número do retângulo A com o número do retângulo B e escrever o resultado no retângulo C



Exemplo de Algoritmo

▶ Preparar um bolo: receita = um algoritmo em que cada instrução é um passo a ser seguido para que o prato fique pronto com sucesso

1. Bata 4 claras em neve
2. Adicione 2 xícaras de açúcar
3. Adicione 2 colheres de farinha de trigo, 4 gemas, uma colher de fermento e duas colheres de chocolate
4. Bata por 3 minutos
5. Unte uma assadeira com margarina e farinha de trigo
6. Coloque a massa do bolo na assadeira
7. Coloque o bolo para assar por 20 minutos

Exemplo de Algoritmo

- ▶ Trocar o um pneu furado: uma rotina para realizar essa tarefa
- 1. Verifica qual pneu está furado
- 2. Posicionar o macaco para levantar o carro
- 3. Pegar o estepe
- 4. Soltar os parafusos
- 5. Substituir o pneu furado
- 6. Recolocar os parafusos
- 7. Descer o carro
- 8. Guardar o macaco e o pneu furado

Problema

- ▶ Ordenar os números abaixo em ordem crescente
- ▶ Escrevam um algoritmo que executa a ordenação

6	8	1	7	3
---	---	---	---	---



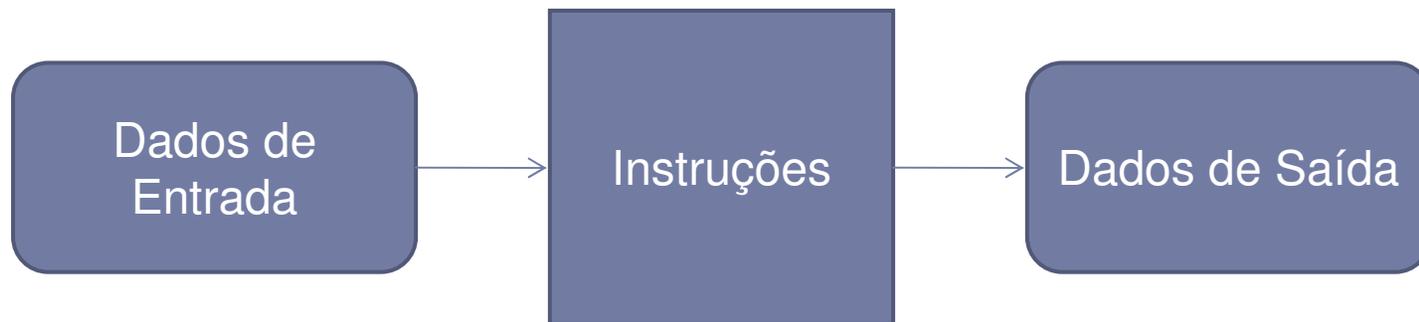
Possível Solução

- ▶ Comparar números um a um e ir fazendo trocas de posição, até que todos estejam ordenados



Computador?

- ▶ E se quisermos que o computador resolva este problema para nós?



- ▶ Em certos casos específicos, os dados de entrada e/ou de saída podem ser ignorados (desnecessários).



Especificando...

- ▶ **Dados de Entrada:**
 - ▶ número de valores a serem ordenados
 - ▶ lista de valores a serem ordenados
- ▶ **Instruções**
 - ▶ seqüência de passos necessários para resolver o problema
- ▶ **Dados de saída:**
 - ▶ lista de valores ordenados



Portanto...

- ▶ O computador não faz nada totalmente sozinho
- ▶ Ele precisa (a princípio) que alguém diga para ele quais os passos que precisam ser executados
 - ▶ Ele precisa que ditemos a ele o ALGORITMO para resolver o problema!
- ▶ Mas... computador não entende qualquer instrução
- ▶ Ele entende apenas um conjunto fixo de instruções
- ▶ Essas instruções precisam ser usadas para resolver qualquer problema



Algoritmo para Computador: PROGRAMA

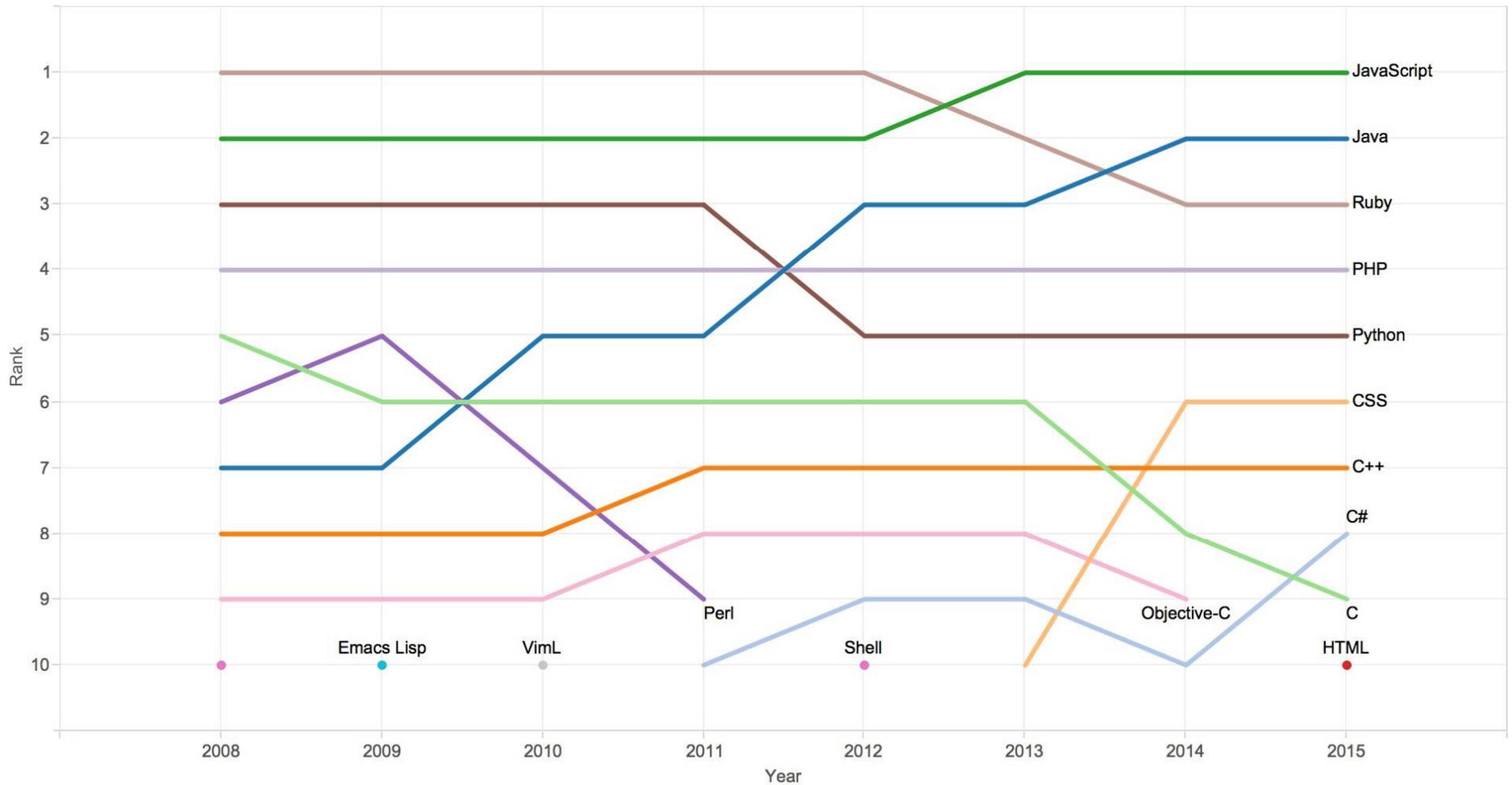
- ▶ Programa = Algoritmo escrito em uma linguagem que o computador entende
 - ▶ linguagens de programação: C, Pascal, COBOL, Fortran, Basic, Java, Perl entre outras
- ▶ Interpretado e executado por um computador
- ▶ Interpretação rigorosa, exata, do computador

- ▶ **Escrita do algoritmo na linguagem de programação tem que seguir regras mais rigorosas... Veremos isso na aula que vem**



Linguagens de Programação Mais Usadas

Rank of top languages on GitHub.com over time



Source: GitHub.com

Representação de Algoritmos

▶ Pseudocódigo

- ▶ Facilita **descrever** o algoritmo **antes** de passá-lo para uma linguagem de programação
- ▶ Intermediária entre **linguagem natural** e linguagem de programação
- ▶ Pseudocódigo = “código falso” ou “quase código”

▶ Fluxograma

- ▶ Forma universal de representação
- ▶ Utiliza figuras geométricas para ilustrar passos a serem seguidos para a resolução de problemas

Pseudocódigo

- ▶ Descrição do algoritmo, menos rigorosa que na linguagem de programação (código fonte)
- ▶ Fácil de entender e fácil de codificar depois (para humanos)
- ▶ Independente da linguagem de programação
- ▶ Simples e objetivo

Técnica:

- ▶ Um verbo por frase
- ▶ Não escrever “para informatas”
- ▶ Frases curtas e simples
- ▶ Ser objetivo
- ▶ Usar palavras sem duplo sentido

Escrevendo Pseudocódigo

- ▶ **FASES para desenvolver o algoritmo**

- ▶ Determinar o problema, defini-lo bem
- ▶ Dividir a solução nas três fases



- ▶ **Exemplo:**

- ▶ **Problema:** calcular a média de quatro números
- ▶ Dados de **entrada:** os números, N1, N2, N3 e N4
- ▶ **Processamento:** somar os quatro números e dividir a soma por 4

$$\frac{N1 + N2 + N3 + N4}{4}$$

- ▶ Dados de **saída:** a média final

Escrevendo Pseudocódigos

$$\frac{N1 + N2 + N3 + N4}{4}$$

- ▶ Algoritmo
 - ▶ Receber o primeiro número
 - ▶ Receber o segundo número
 - ▶ Receber o terceiro número
 - ▶ Receber o quarto número
 - ▶ Somar todos os números
 - ▶ Dividir a soma por 4
 - ▶ Mostrar o resultado da divisão

Fluxograma

- ▶ Representação gráfica padronizada dos passos de um algoritmo
- ▶ Principais símbolos



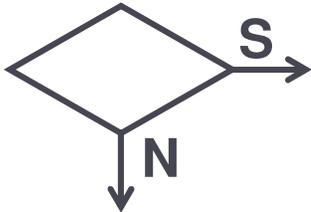
Terminal: INÍCIO ou FIM do algoritmo



Processamento (ação, operação)



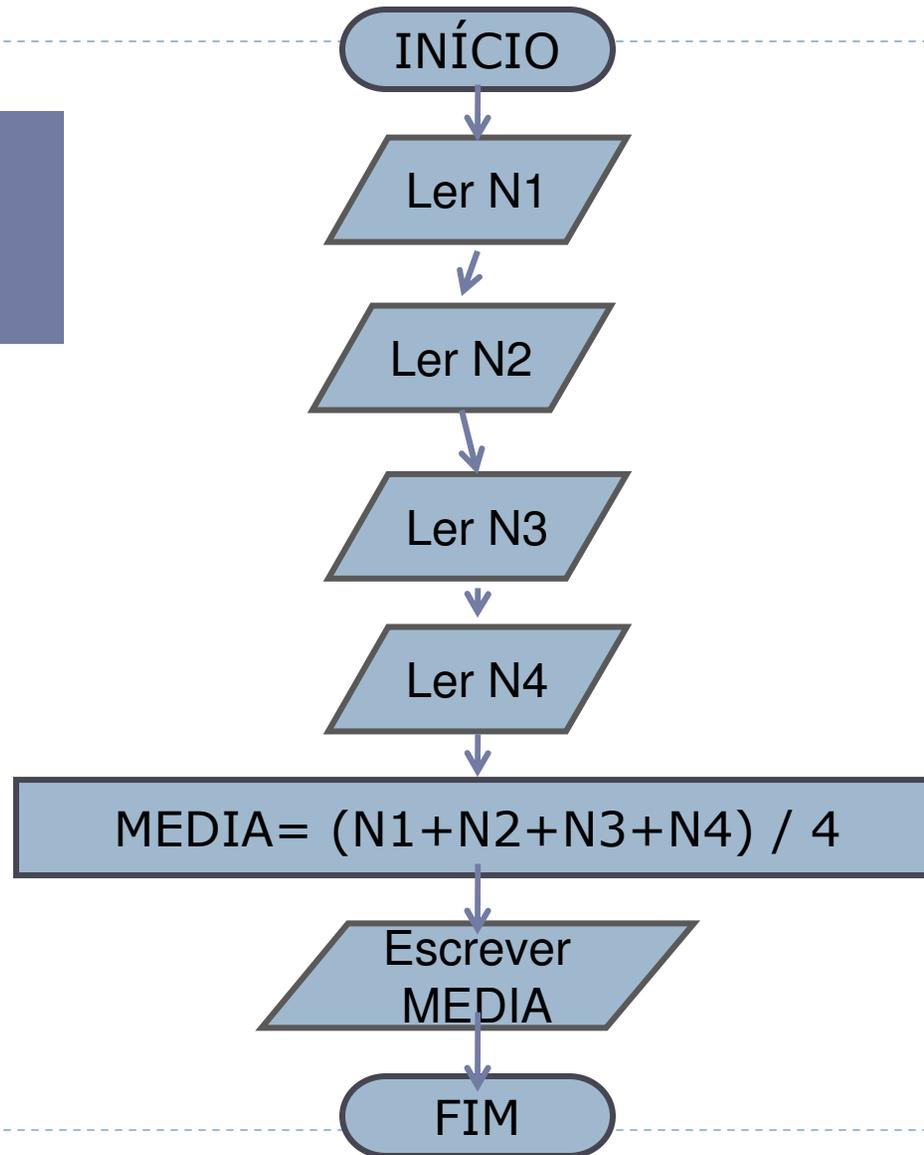
Entrada e saída de dados



Decisão (teste)

Fluxograma

$$\frac{N1 + N2 + N3 + N4}{4}$$



Exercícios

1 - Identifique os dados de entrada, processamento e saída no algoritmo abaixo

- ▶ Receba código da peça
- ▶ Receba valor da peça
- ▶ Receba Quantidade de peças (Qtde)
- ▶ Calcule o valor total da peça (Qtde * Valor)
- ▶ Mostre o código da peça e seu valor total

2 - Faça um algoritmo para “Calcular o estoque médio de uma peça”, sendo que

estoque médio = (quantidade mínima + quantidade máxima) / 2

Vamos praticar no computador?

- ▶ Visual Studio: <https://visualstudio.com/en-us/downloads/download-visual-studio-vs.aspx>
- ▶ Objetivo: Fazer o computador dizer “alô!” a você!

- ▶ Editor de texto (com suporte a C e outras linguagens):
- ▶ Sublime Text 2



Ainda não entendi como o
computador entra nesta história...

de algoritmos

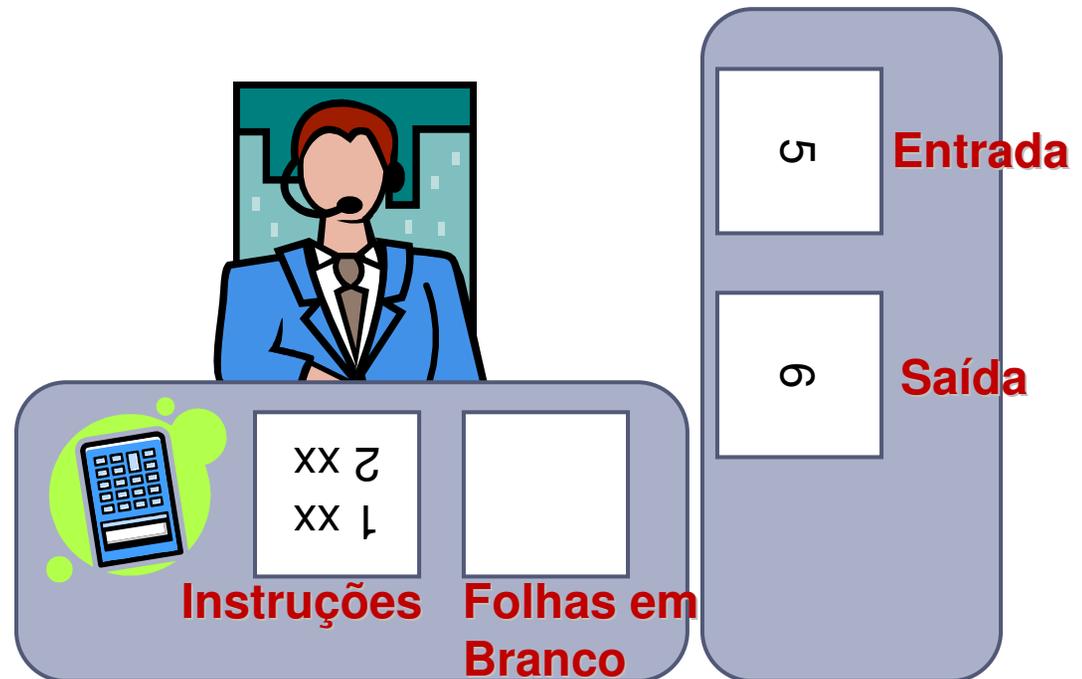
Modelo de um computador

- ▶ Para entender como um computador funciona, vamos pensar num “modelo” que seja de fácil entendimento

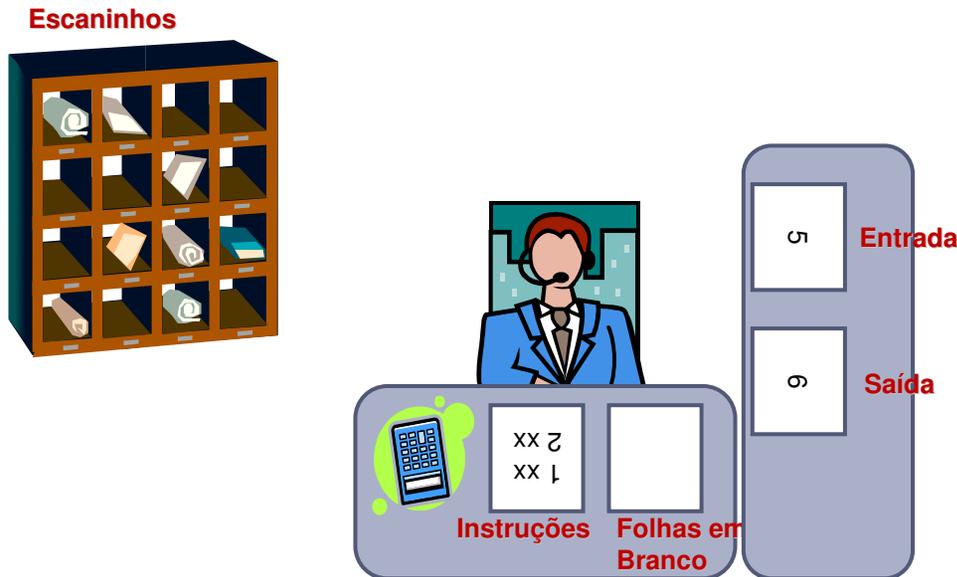


Modelo da secretária

Escaninhos



Modelo da secretária



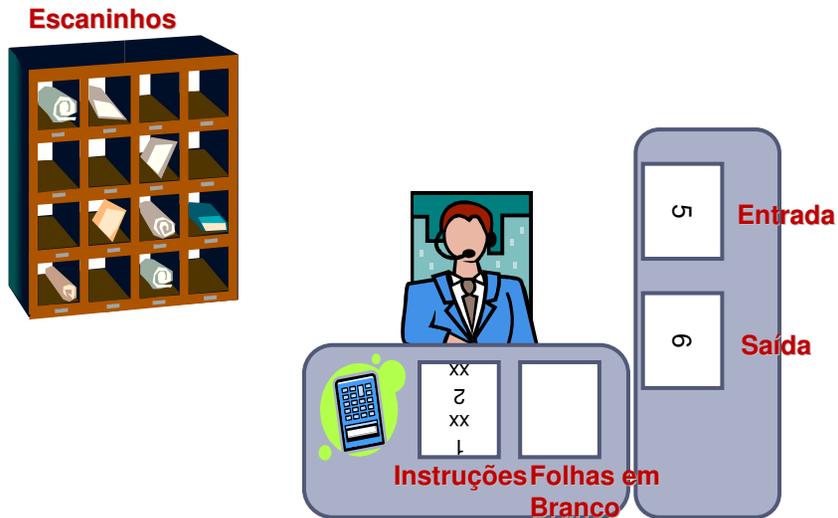
- ▶ Secretária conhece um conjunto pequeno de instruções
- ▶ Ela apenas segue as instruções
- ▶ Escaninhos têm etiqueta com “rótulo”
- ▶ No fim do dia, o *boy* passa e limpa os escaninhos

Conjunto de Instruções que a secretária conhece

- ▶ Pegar um valor (de um escaninho ou da caixa de entrada)
- ▶ Copiar um valor (para um escaninho ou para a caixa de saída)
- ▶ Calcular (somar, subtrair, multiplicar, dividir)
- ▶ Avaliar (expressão lógica) → resultado VERDADEIRO ou FALSO



Exemplo 1



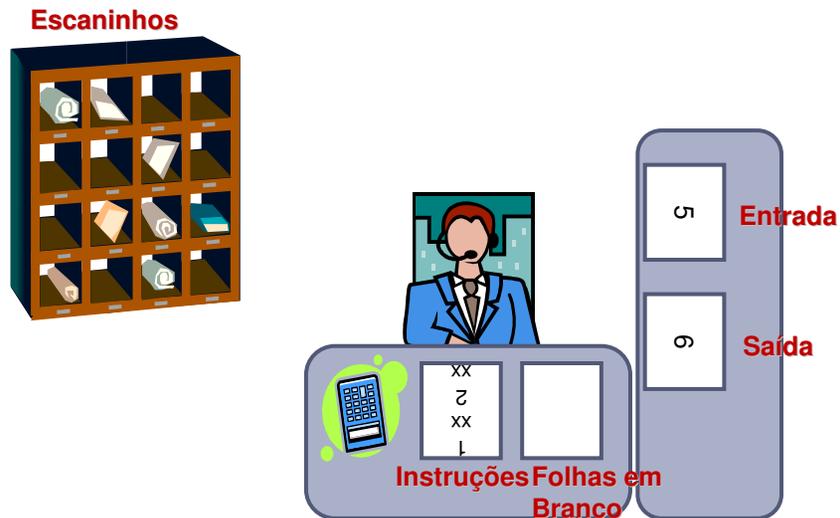
1. Pegue um valor e coloque no escaninho A
2. Pegue um valor e coloque no escaninho B
3. Some o valor contido no escaninho A com o valor contido no escaninho B e coloque o resultado no escaninho SOMA
4. Copie o valor do escaninho SOMA para a caixa de saída.

Instrução “AVALIE”

- ▶ A instrução “avale” avalia uma expressão lógica – o resultado é verdadeiro ou falso
 - ▶ Avalie $2 = 3$?
 - ▶ Avalie $10 > 5$?
- ▶ Conectores lógicos “e” e “ou”
 - ▶ “e” – todos os itens avaliados devem ser verdadeiros para a expressão ser verdadeira
 - ▶ “ou” – basta que um dos itens avaliados seja verdadeiro para que a expressão seja verdadeira
- ▶ Avalie $10 > 5$ ou $2 = 3$?
- ▶ Avalie $10 > 5$ e $2 = 3$?



Exemplo 2



1. Pegue um valor e coloque no escaninho A
2. Pegue um valor e coloque no escaninho B
3. Avalie $A > B$ e coloque o resultado no escaninho R
4. Copie o valor do escaninho R para a caixa de saída.



Modelo da secretária

- ▶ Dá uma noção clara de como o computador funciona
- ▶ Secretária é a CPU do computador (quem executa as instruções)
- ▶ Instruções são os programas
- ▶ Escaninhos são as posições na memória RAM do computador
- ▶ Caixa de Entrada é o teclado ou dados em um arquivo
- ▶ Caixa de Saída é o monitor ou a impressora
- ▶ O boy no fim do dia esvazia o escaninho: Memória RAM do computador é VOLÁTIL



Ficou mais claro?

Agora podemos voltar aos algoritmos...

Elementos de um algoritmo

- ▶ Constantes
- ▶ Variáveis
- ▶ Atribuição
- ▶ Operadores

- ▶ Mais complexos:
 - ▶ Funções
 - ▶ Procedimentos
- ▶ Mais complexos ainda:
 - ▶ Classes / objetos

Constante

- ▶ Representa um valor fixo na memória inicializado junto ao programa (com validade até o término do mesmo), em C, temos:
 - ▶ `#define c 10`
 - ▶ Também pode ser *inicializado* durante o programa
 - ▶ `const int c = 10;`



Variável

- ▶ Representa uma **posição na memória**, onde pode ser armazenado um dado (um ESCANINHO!)
- ▶ Possui um nome e um valor
- ▶ Durante a execução do algoritmo, pode ter seu valor alterado (seu valor pode variar)
- ▶ Mudanças no valor das variáveis:
 - ▶ Por entrada de dados (“Ler N1”)
 - ▶ Por atribuição (“MEDIA = 6”)

Variável: exemplo sequencial

“Calcular a média de quatro números”

Pseudocódigo:

- ▶ Ler N1
- ▶ Ler N2
- ▶ Ler N3
- ▶ Ler N4
- ▶ $MEDIA = (N1+N2+N3+N4) / 4$
- ▶ Mostrar MEDIA

Variável: exemplo seqüencial

“Calcular a média de quatro números”

▶ PSEUDOCÓDIGO:

- ▶ Ler N1
- ▶ Ler N2
- ▶ Ler N3
- ▶ Ler N4

VARIÁVEIS:
mais clareza no pseudocódigo

- ▶ $MEDIA = (N1 + N2 + N3 + N4) / 4$
- ▶ Mostrar MEDIA

VARIÁVEL

Atribuição

- ▶ Atribui o valor da direita à variável da esquerda
- ▶ O valor pode ser uma constante, uma variável ou uma expressão

$$\text{MEDIA} = (\text{N1} + \text{N2} + \text{N3} + \text{N4}) / 4$$

(Lê-se média recebe N1+...)

- ▶ Neste caso, estamos atribuindo o resultado da fórmula à variável média, ou seja, a variável média está recebendo como valor o resultado da fórmula
- ▶ Outros Exemplos
 - a = 3;
 - a = x;

Operadores

- ▶ São operações básicas em processamento de dados
- ▶ Usados para **incrementar, decrementar, comparar e avaliar dados**
- ▶ Tipos:
 - ▶ Aritméticos (+, -, *, /, ** ou ^)
 - ▶ Resultados numéricos
 - ▶ Relacionais (>, <, >=, <=, =, <> ou #)
 - ▶ Resultados lógicos (V ou F)
 - ▶ Lógicos (e, ou, não)
 - ▶ Combinam resultados lógicos

Operadores Aritméticos

OPERAÇÃO	SÍMBOLO
Adição	+
Subtração	-
Multiplicação	*
Divisão	/
Exponenciação	** ou pow(n1,n2)

Hierarquia das Ops. Aritméticas

Parênteses

Exponenciação

Multiplicação ou Divisão

Adição ou Subtração

▶ Exemplos:

▶ $TOTAL = PRECO * QUANTIDADE$

▶ $1 + 7 * 2 ** 2 - 1$

▶ $3 * (1-2) + 4 * 2$

▶ $MEDIA = (N1 + N2 + N3 + N4)/4$

Exercícios

1. Faça um algoritmo que tendo como dados de entrada a altura de uma pessoa, ***h***, calcule seu ***peso ideal***, utilizando a seguinte fórmula

$$\text{peso ideal} = (72.7 * h) - 58$$

2. Faça um algoritmo que receba como entrada uma determinada temperatura em **graus Celsius** e mostre a temperatura em graus **Fahrenheit**

OBS: Fahrenheit = (9/5)*(Celsius) + 32

3. Faça um algoritmo que tendo como entrada o ***total vendido*** por um funcionário no mês, mostre sua ***comissão*** e o seu ***salário bruto*** neste mês. Sendo que o seu ***salário base*** é R\$1.200,00 e sua comissão é de ***10% sobre o total vendido*** por ele.

Operadores Relacionais

DESCRIÇÃO	SÍMBOLO
Igual a	=
Diferente de	<> ou # ou !=
Maior que	>
Menor que	<
Maior ou igual a	>=
Menor ou igual a	<=

Exemplo:

Suponha duas variáveis: A e B

A = 5

B = 3

- ▶ Muito usados para tomar decisões nos algoritmos
- ▶ Usados para: testes, comparações, que resultam em valores lógicos (verdadeiro ou falso)

ExpressãoA	Resultado
A = B	FALSO
A != B	VERDADEIRO
A > B	VERDADEIRO
A < B	FALSO
A >= B	VERDADEIRO
A <= B	FALSO

Exercícios

1. Tendo as variáveis SALARIO, IR e SALLIQ, e considerando os valores abaixo. Informe se as expressões são **verdadeiras** ou **falsas**

SALARIO	IR	SALLIQ	EXPRESSAO	V ou F
100,00	0,00	100,00	(SALLIQ >= 100)	
200,00	10,00	190,00	(SALLIQ < 190,00)	
300,00	15,00	285,00	SALLIQ = SALARIO - IR	

2. Sabendo que $A=3$, $B=7$ e $C=4$, informe se as expressões abaixo são **verdadeiras** ou **falsas**.
- $(A+C) > B$ ()
 - $B \geq (A + 2)$ ()
 - $C = (B - A)$ ()
 - $(B + A) \leq C$ ()
 - $(C+A) > B$ ()

Operadores Lógicos

A	B	A AND B
T	T	T
T	F	F
F	T	F
F	F	F

A	B	A OR B
T	T	T
T	F	T
F	T	T
F	F	F

A	NOT A
T	F
F	T

Hierarquia dos Ops. Lógicos:
Parênteses
NOT
AND
OR

- ▶ Combinam resultados lógicos
- ▶ Geram novos valores lógicos (T ou F)
- ▶ A “tabela-verdade” mostra todos os valores possíveis de se obter com operadores lógicos

LEGENDA:

T = VERDADEIRO

F = FALSO

AND = E

OR = OU

NOT = NÃO

Operações Lógicas

- ▶ Operadores relacionais + operadores lógicos = operações lógicas
- ▶ Produzem resultados lógicos (T/F)
- ▶ Para $A = 5$, $B = 8$ e $C = 1$:
 - ▶ $(A = B) \text{ AND } (B > C)$
 - ▶ $(A <> B) \text{ OR } (B < C)$
 - ▶ $\text{NOT } (A > B)$
 - ▶ $(A < B) \text{ AND } (B > C)$
 - ▶ $(A \geq B) \text{ OR } (B = C)$
 - ▶ $\text{NOT } (A \leq B)$
- ▶ São usadas em decisões nos algoritmos...

Operações Lógicas

- ▶ Operadores relacionais + operadores lógicos = operações lógicas
- ▶ Produzem resultados lógicos (T/F)
- ▶ Para $A = 5$, $B = 8$ e $C = 1$:
 - ▶ $(A = B)$ **AND** $(B > C)$ F (V AND F)
 - ▶ $(A <> B)$ **OR** $(B < C)$ V (V OR F)
 - ▶ **NOT** $(A > B)$ V (NOT F)
 - ▶ $(A < B)$ **AND** $(B > C)$ V (V AND V)
 - ▶ $(A >= B)$ **OR** $(B = C)$ F (F OR F)
 - ▶ **NOT** $(A <= B)$ F (NOT V)
- ▶ São usadas em decisões nos algoritmos...

Exercício

Considere a seguinte atribuição de valores para as variáveis: $A=3$, $B=4$, $C=8$. Avalie as expressões indicando o resultado final: T / F

- ▶ $A > 3 \text{ AND } C = 8$
- ▶ $A <> 2 \text{ OR } B \leq 5$
- ▶ $A = 3 \text{ OR } B \geq 2 \text{ AND } C = 8$
- ▶ $A = 3 \text{ AND NOT } B \leq 4 \text{ AND } C = 8$
- ▶ $A <> 8 \text{ OR } B = 4 \text{ AND } C > 2$
- ▶ $B > A \text{ AND } C <> A$
- ▶ $A > B \text{ OR } B < 5$
- ▶ $A <> B \text{ AND } B = C$
- ▶ $C > 2 \text{ OR } A < B$
- ▶ $A > B \text{ OR } B > A \text{ AND } C <> B$

Exercício

Sabendo que $A=5$, $B=4$ e $C=3$ e $D=6$, informe se as expressões abaixo são **verdadeiras** ou **falsas**.

- ▶ $(A > C)$ **AND** $(C \leq D)$
- ▶ $(A+B) > 10$ **OR** $(A+B) = (C+D)$
- ▶ $(A \geq C)$ **AND** $(D \geq C)$

Exercício

Sabe-se que o uso incorreto da precedência de operadores ocasiona erros. Pensando nisso, determine o resultado das expressões a seguir (valores: $A = 8$, $B = 5$, $C = -4$, $D = 2$)

- ▶ $\Delta = B - 4 * A * C$
- ▶ $J = \text{"Hoje"} \langle \rangle \text{"HOJE"}$
- ▶ $\text{Media} = (A + B + C + D) / 4$
- ▶ $\text{Media} = A + B + C + D / 4$
- ▶ $\text{Resultado} = A + B - 10 * C$
- ▶ $Y = A > 8 \text{ E } B + C > D$
- ▶ $Y = A > 3 * 2 \text{ OU } B + C \langle \rangle D$

Ciclo de Vida de um Programa

- ▶ Especificação de requisitos (Entradas/Saídas/ o que o programa deve fazer)
- ▶ Algoritmo (solução)
- ▶ Testes
- ▶ Programação
- ▶ Testes
- ▶ Manutenção



Teste

- ▶ **Todo algoritmo deve ser testado**
 - ▶ Usar dados e resultados previamente calculados
 - ▶ Seguir precisamente as instruções do algoritmo
 - ▶ Verificar se o procedimento está correto
- ▶ Exemplo: Fazer teste de mesa (china) para o algoritmo da média (nota máxima = 10)

P1	P2	P3	P4	Média

Meu primeiro programa em C

...

Integrated Development Environment (IDE) e Compiladores

- ▶ *No Visual Studio, crie um novo projeto como no tutorial:*
 - ▶ https://youtu.be/VKds2loxc_U
 - ▶ <https://www.youtube.com/watch?v=u60ABTDYyNc>

- ▶ Ou, inicie o Dev-C++ pelo menu
- ▶ Crie um novo arquivo, com o comando *File, New Source File*
- ▶ Edite o programa da página seguinte

Visual Studio Code

Entrada (1,1 x) IADES - Inst x www2.ic.ufi x Google Tra x Nothing Bu x Spotify Play x Visual Studi x Downloads x Download x visual studi x most used x read value x Ambiente d x

Visual Studio Code Docs Updates Connect Gallery FAQ Search

VS Code Beta is available. Check out the [new features](#) and [update](#) it now.

Code focused development. Redefined.

Build and debug modern web and cloud applications. Code is free and available on your favorite platform - Linux, Mac OSX, and Windows.

[Download for Windows](#) [Download for Linux x64](#) [Download for OS X](#)

32bit version

By downloading and using Visual Studio Code, you agree to the [license terms](#) and [privacy statement](#) for Visual Studio Code. When this tool crashes, we automatically collect crash dumps so we can figure out what went wrong. If you don't want to send us crash dumps, go [here](#) to learn how to disable them.

Pesquisar na Web e no Windows

20:38 30/11/2015

→ Vantagem: Gratuito e multiplataforma

IDE e Compiladores

- ▶ Visual Studio != Visual Studio Code
 - ▶ O visual studio community é gratuito (apenas Windows):
 - ▶ <https://www.visualstudio.com/pt-br/products/visual-studio-community-vs.aspx>
 - ▶ Diversos compiladores e IDEs podem ser utilizados, o requerimento é compilar código na linguagem C/C++.
 - ▶ Também é possível utilizar o Visual Studio em outros sistemas operacionais, utilizando máquinas virtuais.
-

Meu Primeiro Programa em C

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

```
int main()
{
    printf("Alô! \n");
    system("pause");
}
```



Como Fazer o Computador Executá-lo?

1. No dev-C++:

1. Salve o programa com o nome **exemplo.c**.
2. Compile o programa com o comando **Executar, Compilar** ou com a tecla **Ctrl-F9**
3. Se houver algum erro de sintaxe, aparece uma ou mais mensagens no rodapé da janela. Neste caso, corrija o programa e repita.
4. Se não houver erros, execute o programa com o comando **Executar, Executar** ou com a tecla **Ctrl-F10**



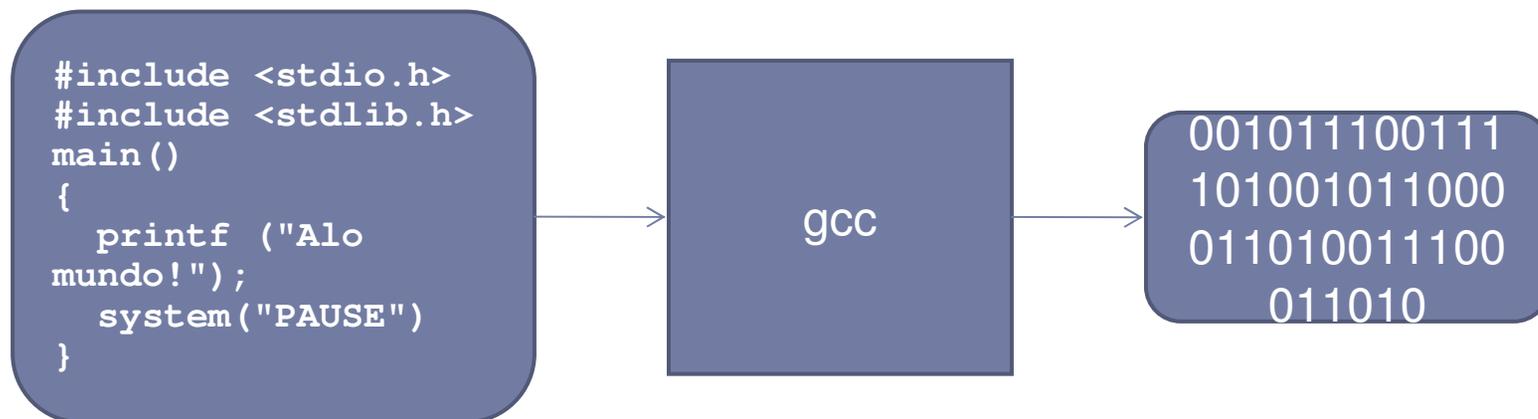
Por que Preciso Compilar?

- ▶ Computador só “entende” zeros e uns...



Por que Preciso Compilar?

- ▶ Computador só “entende” zeros e uns...



Exercício Prático

Faça um programa que imprime o seu nome na tela e aguarda o acionamento da tecla Enter para terminar.

Dicas:

- ▶ Para imprimir texto direto na tela, sem que ele seja valor de alguma variável, usar aspas.

```
printf ("Maria");
```

- ▶ Isso é diferente de imprimir o valor da variável A (neste caso não se usa aspas)

```
A = "Maria";
```

```
printf (A);
```



Na próxima aula...

- ▶ Veremos alguns comandos da linguagem de programação C

- ▶ Até lá.....

