

# Programação de Computadores

Prof. Msc. Carlos Queiroz

Semestre 2013.1

# Agenda

- Apresentação
- Ementa da Disciplina
- Avaliação
- Bibliografia
- Introdução

# Apresentação

- Carlos Alberto
- Graduação
- Especialização
- Mestrado
- Histórico Profissional

# Apresentação (Alunos)

- Algoritmo para apresentação:

Faça {

- Diga o seu Nome;
- Responda: Trabalha ou trabalhou? Onde?
- Responda: Já possui algum conhecimento relacionado a disciplina?
- Responda: O que achou da disciplina de algoritmos?
- Indique um colega para se apresentar;

}

Enquanto (tiverem alunos que não se apresentaram )

# Ementa

## ■ 1 INTRODUÇÃO

- 1.1 Conceito de programa de computador
- 1.2 Conceito de programação estruturada
- 1.3 Conceito de algoritmo
- 1.4 Estrutura de um programa

## ■ 2 PROGRAMAÇÃO ESTRUTURADA

- 2.1 Definição
- 2.2 Elementos

# Ementa (continuação)

- 3 INTRODUÇÃO À LINGUAGEM C
  - 3.1 Estrutura geral de um programa em “C”
  - 3.2 Entrada e saída de dados
  - 3.3 Variáveis, constantes e expressões
  - 3.4 Tipos de dados
  - 3.5 Operadores aritméticos
  - 3.6 Operadores lógicos
  - 3.7 Estruturas de desvio condicional
  - 3.8 Estruturas de repetição
  - 3.9 Funções
  - 3.10 Tipos de dados estruturados (Vetor, Matriz e String)
- Concepção e implementação de programas

# Avaliação

## ■ Trabalhos

- Listas de Exercícios (Duplas)
- Seguir os padrões de entrega que serão informados

## ■ Provas

- NP1 (Prova + Trabalhos)
  - Prova => 70% da nota
  - Trabalho => 30% da nota
- NP2 (Prova + Trabalhos)
  - Prova => 70% da nota
  - Trabalho => 30% da nota

# Avaliação

- Segunda Chamada
  - Deve ser solicitada na secretaria
  - Prova única
  - Todo o conteúdo
  
- $\text{Nota} = (\text{Np1} + \text{Np2}) / 2 \Rightarrow \text{Média}$ 
  - Se  $\text{média} > 7,0$  o aluno estará aprovado
  - Se  $4,0 < \text{média} < 7,0$  o aluno estará habilitado a fazer prova final
  - Se  $\text{média} < 4,0$  o aluno estará reprovado



# Avaliação

- Provas

- Prova Final

- Para alunos que ficaram com média superior a 4,0 e inferior a 7,0

- Todo o conteúdo

- A média final é obtida através do cálculo

- $MF = (Média + Nota da prova final) / 2$

- Se  $MF > 5,0$  o aluno estará aprovado

- Se  $MF < 5,0$  o aluno estará reprovado

# Bibliografia Básica

- SCHILDT, H., **C Completo e Total**, Editora Makron Books, 3a. ed., 1997.
- MIZRAHI, V. V., **Treinamento em linguagem C**. 2ª edição. Editora Pearson. 2008
- DEITEL, H. e DEITEL, P. **C: Como Programar**. 6ª Edição. Editora Pearson. 2011.
- SENNE, E. L. **Primeiro Curso de Programação em C**. 3ª Edição. Editora Visual Books. 2009.



**1. Desconfiança**  
Eu posso fazer  
isso?



**2. Excitação**  
Eu posso fazer  
isso!



**3. Espanto**  
Como vou fazer isso?



**4. Entusiasmo**  
Eu consegui fazer  
isso!



**5. Amor**  
Sou um excelente  
programador!



**6. Desilusão**  
O código não está  
funcionando



**7. Sujo**  
Essa lógica não  
vai funcionar!?



**8. Horror**  
Um bug pior! Que droga de computador  
%#&%!#&@\*#&#



**9. Fúria**



**10. Frustração**  
Não está funcionando  
como deveria



**11. O fim**  
Terapia Ocupacional

# Aula 1 - Introdução

- Agenda
  - Conceito de Algoritmo
  - Conceito de programa de computador

# Conceito de Algoritmo

“Algoritmo é uma sequência de passos que visa atingir um objetivo bem definido”  
(Forbellone, 1999)

“Algoritmo é a descrição de uma sequência de passos que deve ser seguida para realização de uma tarefa”  
(Ascencio, 1999)

# Conceito de Algoritmo (cont.)

## ■ Exemplo

### ■ Trocar uma lâmpada...

- ① Pegar Lâmpada nova
- ② Pegar Escada
- ③ Posicionar escada embaixo da lâmpada queimada
- ④ Subir na escada com a lâmpada nova na mão
- ⑤ Retirar lâmpada queimada
- ⑥ Colocar lâmpada nova
- ⑦ Descer da escada
- ⑧ Testar no interruptor
- ⑨ Guardar a escada
- ⑩ Descartar Lâmpada velha



# Conceito de Algoritmo (cont.)

- Computadores podem ajudar na realização de tarefas mais complexas
  - Problemas envolvendo um grande volume de dados
  - Problemas que exijam um grande número de operações
  - Problemas que exijam precisão
- Características de um bom Algoritmo
  - Finitude
  - Exatidão
  - Entradas e Saídas determinadas
  - Efetividade

# Conceito de Algoritmo (cont.)

- Representação

- Linguagem Natural

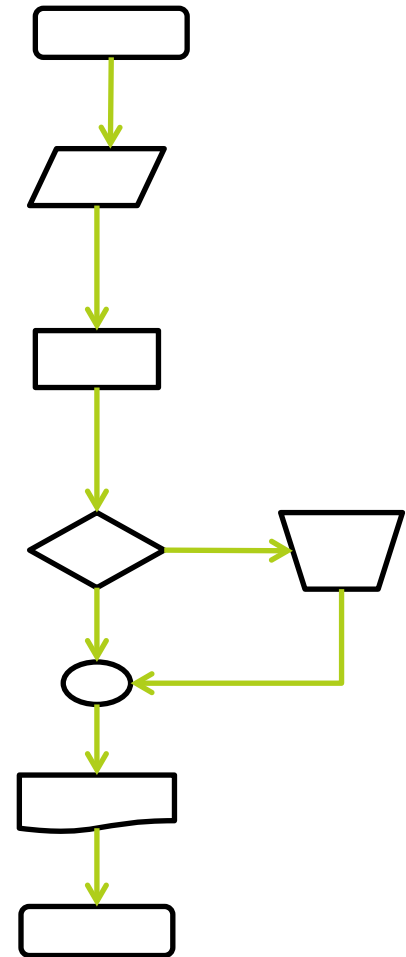
- Descrição textual da tarefa em linguagem natural

- Forma narrativa



# Conceito de Algoritmo (cont.)







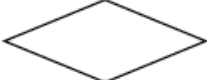


- Representação
  - Fluxograma
    - Representação Gráfica estruturada e simplificada
    - Ilustra o fluxo de operações
    - Sintaxe e Semântica bem definidos
    - Fácil para traduzir para uma linguagem de programação



# Conceito de Algoritmo (cont.)

- A sintaxe de um fluxograma corresponde ao emprego correto de seus Elementos
  - Cada instrução possui um símbolo gráfico específico
  - Cada símbolo admite um conjunto de expressões próprias
  - Sub-rotinas pré-definidas podem ser utilizadas nas expressões
- A semântica de um fluxograma corresponde ao significado dos símbolos utilizados
  - Cada instrução deve ser descrita de forma clara, sem ambiguidade

# Conceito de Algoritmo (cont.)

Símbolo	Nome	Função
	terminador	Determina o início ou o fim do processamento.
	conector	Indica ligações com outras partes de um fluxograma.
	fluxo	Indica o sentido da execução do algoritmo.
	processo	Realiza o cálculo de expressões e/ou a execução de funções.
	atribuição	Armazenamento temporário do resultado de um processamento.
	sub-rotina	Processamento de funções ou chamada a procedimentos.
	decisão	Avaliação de uma expressão relacional ou lógica.
	leitura	Entrada manual de dados, em tempo de execução.
	exibição	Apresentação dos resultados do processamento.

# Conceito de Algoritmo (cont.)

- Representação
  - Pseudo-código

```
ALGORITMO  
DECLARE N1, N2, M NUMÉRICO  
ESCREVA "Digite dois números"  
LEIA N1, N2  
M ← N1 * N2  
ESCREVA "Multiplicação = ", M  
FIM_ALGORITMO.
```

# Conceito de Programa

- Programa de Computador
  - Algoritmo escrito em uma linguagem de programação



# Linguagem de Máquina

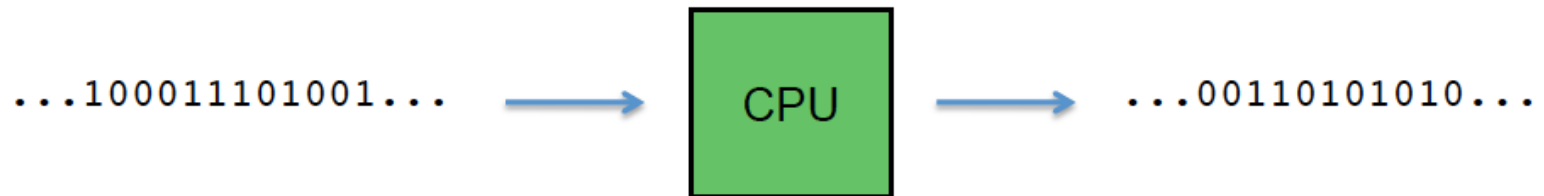
- Um processador executa apenas instruções escritas em linguagem de máquina (dígitos binários)
- Cada instrução é composta de duas partes

Comando      e      operando

1000

11101001

- A Linguagem de máquina é a mais rápida



# Linguagem de Baixo Nível

- São Linguagens que utilizam palavras para definir as instruções para o processador
- Antes de serem executadas são traduzidas para linguagem de máquina

<b>comando</b>	<b>operando</b>	<b>significado</b>
LD	A	load A
MPY	5	multiply by 5
STO	B	store in B

# Linguagem de Alto Nível

- São Linguagens de programação que permitem escrever programas usando um vocabulário semelhante a maneira natural de se expressar o problema que se deseja resolver
- Exemplos
  - C, C++, Java, C#, Pascal, Python, Objective-C e várias outras...

```
read(A);  
B = A + 5;
```



# Compilador

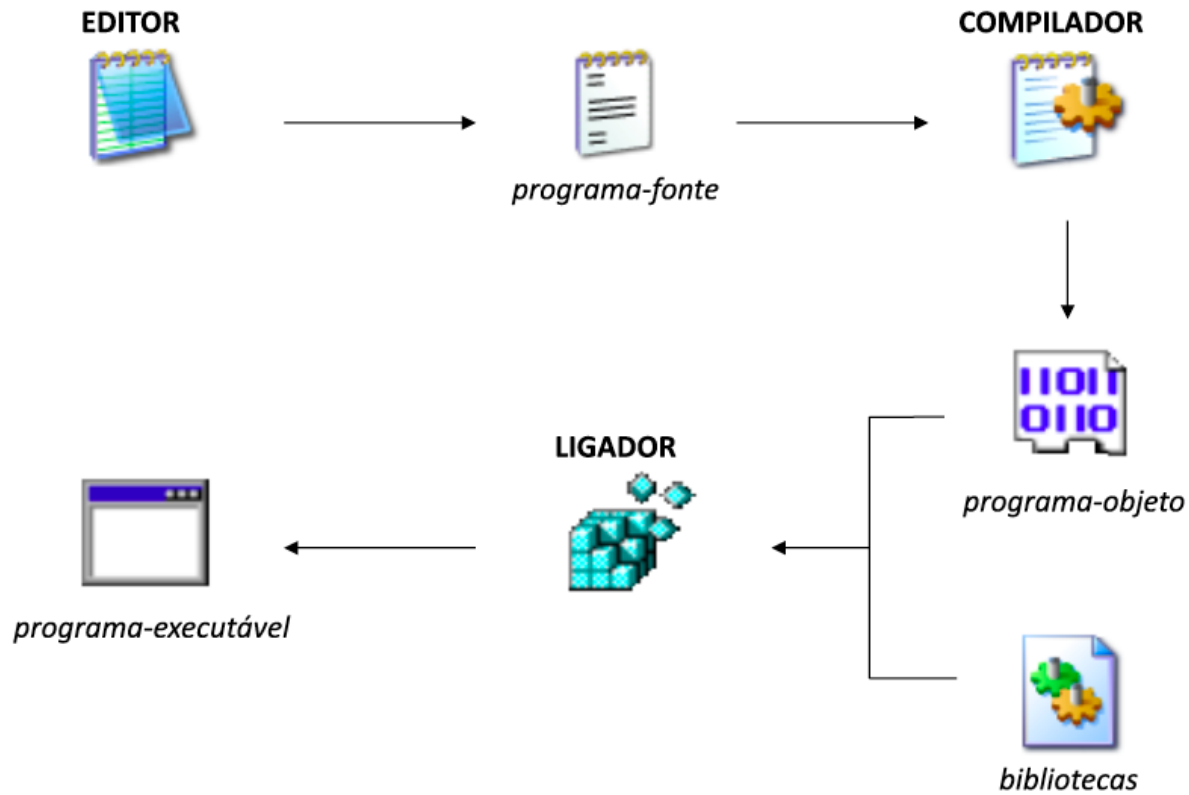
- Programas que traduzem programas escritos em linguagem de programação para programas equivalentes escritos em linguagem de máquina
- O primeiro é chamado de programa fonte, enquanto que o segundo é chamado de programa objeto
- A compilação também verifica se o programa obedece à sintaxe da linguagem, ou seja, se as instruções foram escritas corretamente



# Linker

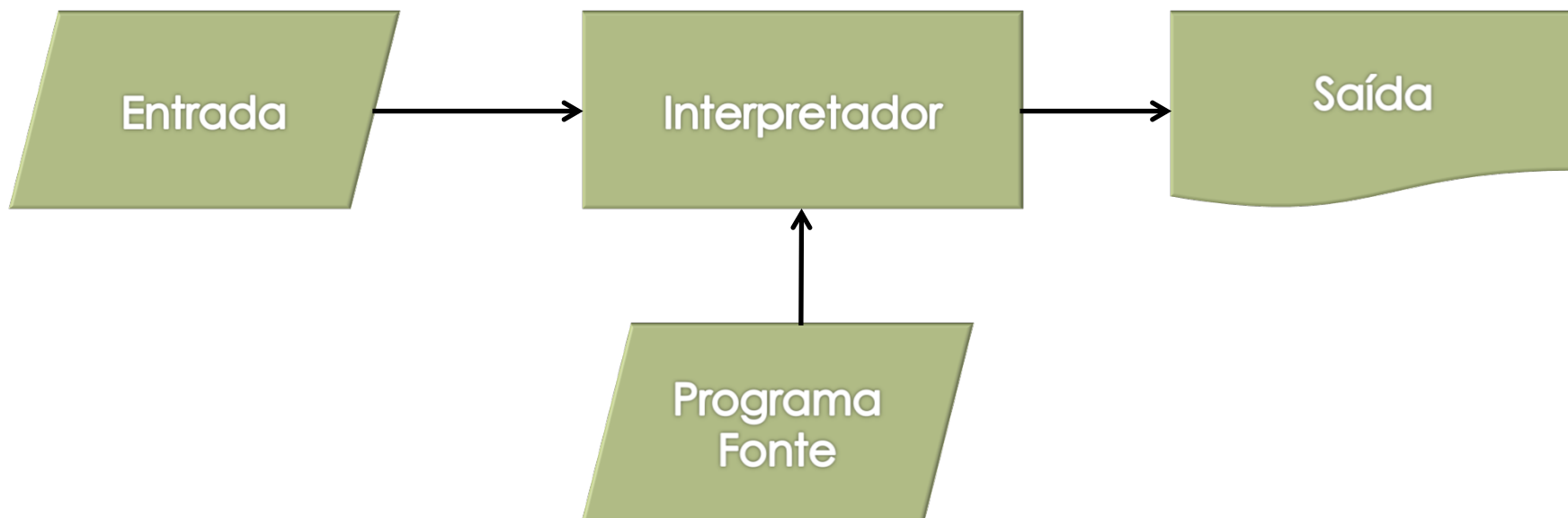
- O programa-objeto nem sempre está pronto para ser executado
- Outros trechos de código precisam ser incluídos
  - Bibliotecas
  - Subprogramas
- Programa executável é montado por um programa chamado link-editor ou linking-loader
- Compilador normalmente chama o link-editor automaticamente

# Compilação



# Interpretador

- A linguagem que iremos estudar não utiliza o conceito de interpretadores
- Simulação de uma “Máquina Virtual”
- O Programa fonte é lido, entendido e as instruções são executadas imediatamente



# Híbridos

- A linguagem que iremos estudar não utiliza o conceito de interpretadores
- Simulação de uma “Máquina Virtual”
- O Programa fonte é lido, entendido e as instruções são executadas imediatamente

