

## SCC 124 - Introdução à Programação para Engenharias

### Programação



Professor: André C. P. L. F. de Carvalho, ICMC-USP  
Pos-doutorando: Isvani Frias-Blanco  
Monitor: Henrique Bonini de Britto Menezes

1

## Aula de hoje

- Comunicação com computadores
- Programação
- Linguagens de programação
- Tradutores
- Algoritmos
- Exemplos

© André de Carvalho - ICMC/USP

2

## Comunicação com computadores

- As pessoas se comunicam utilizando linguagem natural
- Por que não usar para se comunicar com computadores?
- Permitiria escrever programas em português:
  - Ex.: Programa para calcular Folha de pagamento
    - *Calcular quanto pagar a cada funcionário, informar funcionários e alocar recursos*
      - Retorna salário bruto, benefícios, descontos e valor final e realiza pagamento

© André de Carvalho - ICMC/USP

3

## Comunicação com computadores

- Português não é uma boa linguagem para dar comandos de forma precisa
  - Vide leis, regulamentos e bulas
- Linguagem natural apresenta
  - Ambiguidades
  - Redundâncias
  - Metáforas
- **Computador não entende linguagem natural**

© André de Carvalho - ICMC/USP

4

## Comunicação com computadores

- No passado, foi aprovada uma lei na Califórnia proibindo motociclistas de andar sem capacete
  - Logo depois de entrar em vigor, um policial parou um motociclista sem capacete
  - O motociclista sugeriu ao policial ler novamente a lei

© André de Carvalho - ICMC/USP

5

## Comunicação com computadores

- O que pedia a lei:
  - Motociclista deve ter um capacete à prova de quebra
  - O capacete deve estar firmemente apertado
- O guarda não pode multar o motociclista

© André de Carvalho - ICMC/USP

6

## Comunicação com computadores

- O que pedia a lei:
  - Motociclista deve ter um capacete à prova de quebra
  - O capacete deve ser firmemente apertado
- O guarda não pode multar o motociclista
  - Motociclista tinha o capacete, a prova de quebra, firmemente apertado ao seu joelho

## Comunicação com computadores

- Comandos para computadores devem ser precisos (sem ambiguidade) e detalhados
  - Português não pode ser usado para enviar instruções a um computador
- Como seres humanos podem se comunicar com os computadores?
  - Usando uma gramática que evite ambiguidade e redundância
    - Linguagem formal
    - Linguagem de programação

## Linguagens de programação

- Conjunto de comandos e estruturas de dados
  - Informa ao computador os comandos a serem executados para realizar uma tarefa
    - E como os dados devem estar organizados
    - Sem ambiguidades e imprecisões
- Diferentes níveis
  - Linguagens de máquina
  - Linguagens de montagem
  - Linguagens de alto nível

## Linguagens de máquina

- Usadas para escrever programas no início da computação
  - Comandos faziam básicas que podiam ser executadas diretamente na máquina
    - Ex: *001 10110*
  - Qualquer computador entende internamente apenas sua própria linguagem de máquina
    - Definida pelo seu hardware

## Linguagens de máquina

- Estrutura da linguagem reflete a estrutura do hardware
  - Não reflete as necessidades do programador
  - Difícil de escrever programas (programar)
- Ex.: Programa que soma salário base com hora extra

```
011 100110
000 100110
100 100100
```

Cada instrução é constituída de 2 partes:  
código da operação      operando (valor)  
011                              100110

## Linguagem de montagem (*Assembly*)

- Abreviações em inglês para representar operações elementares
- Programas tradutores (*assemblers*)
  - Convertiam programas em linguagem *assembly* para programas em linguagem de máquina
- Ex.:  
LOAD BASE  
ADD EXTRA  
STORE SALARIO

## Linguagem de montagem (*Assembly*)

- Abreviações em inglês para representar operações elementares
- Programas tradutores (*assemblers*)
  - Convertem programas em linguagem *assembly* para programas em linguagem de máquina

■ Ex.:

```
LOAD BASE      011 100110
ADD EXTRA     000 100110
STORE SALARIO 100 100100
```

## Linguagem de alto nível

- Linguagem de montagem requer grande número de instruções
  - Mesmo para programas simples
- Conseqüências:
  - Fácil cometer erros
  - Difícil consertar erros e atualizar programas
  - Grande esforço manual

## Linguagem de alto nível

- Reduz número de comandos
  - Comando simples em ling. de alto nível  $\approx$  conjunto de comandos em ling. de máquina
  - Programas tradutores convertem
    - Código em ling. de alto nível para código em ling. de máquina
      - Código fonte  $\rightarrow$  código objeto
- Ex.:
  - salario = base + extra

```
011 100110
000 100110
100 100100
```

## Linguagens de alto nível

- Existem centenas de linguagens de alto nível
- Por que tantas linguagens?
  - Propósitos diferentes
    - Aplicativos, comerciais, web, programas científicos, jogos, sistemas embarcados...
  - Avanços tecnológicos
  - Interesses comerciais e científicos



## Linguagens de programação

- O que é uma linguagem de programação?
  - Abstração de uma máquina virtual

```
values = [1, 3, 5, 0.1]
total = sum(values)
print ("resultado: ")
print (total)
```



```
00101010101010
10101011111010
11101010101110
00101010101010
...
```

Programação: processo de escrita de programas em uma linguagem de programação

## Programação

*“Programação é a arte de dizer a um outro ser humano o que você deseja que o computador faça”*

Donald Knuth  
The *Art of Computer Programming*

“Um dos 100 livros que criaram um século de ciência”  
American Scientist

## Tradutores

- Computador não entende programa escrito em linguagem de alto nível
  - Necessário usar programas tradutores
    - Traduzem código fonte (ling. de alto nível) em código objeto (ling. de máquina)
  - Tradutores
    - De linguagens de montagem → montadores
    - De linguagens de alto nível → compiladores e/ou interpretadores

## Tradutores

- Compilador
  - Traduz código fonte para um código objeto em linguagem de máquina
    - Que depois será executado
- Interpretador
  - Decodifica unidades básicas do código fonte (ex. comandos) e as executa imediatamente
    - Pode ser computacionalmente ineficiente
      - Por decodificar independentemente trechos semelhantes do código fonte
    - Facilita programação

## Programa pode ser escrito

- Diretamente em uma linguagem de programação ou
- Inicialmente em um algoritmo
  - Pseudocódigo
  - Depois traduzidos para uma linguagem de programação
    - Programa implementa o algoritmo

## Algoritmo

- Primeiro passo para escrever um algoritmo
  - Identificar precisamente qual é o problema a ser resolvido
- Duas questões importantes:
  - O algoritmo funcionará corretamente?
  - Quanto tempo levará a execução do programa correspondente?

## Algoritmo

- Sequência de passos para realizar uma tarefa
  - Tarefa é especificada por suas entradas e saídas desejadas
  - Algoritmo fornece um método para converter as entradas nas saídas desejadas
  - Deve ser preciso e não ambíguo

## Algoritmo

- Procedimento passo a passo para resolver um problema

Pessoas tem inteligência e habilidade racional

Fazem perguntas para esclarecer dúvidas

Computadores não têm Inteligência e bom senso

Devem receber instruções explícitas (algoritmos)

## Algoritmo

- Um algoritmo correto deve possuir 3 qualidades:
  1. Cada passo do algoritmo deve ser uma instrução que possa ser executada
  2. A ordem dos passos deve ser precisamente determinada
  3. O algoritmo deve ter fim

## Algoritmos corretos x incorretos

- Algoritmo é dito correto se para cada possível entrada, produz a saída correta
  - É incorreto quando produz pelo menos uma saída incorreta
  - Mais fácil mostrar que um algoritmo é incorreto do que correto
  - Métodos formais permitem provar corretude de algoritmos

## Recomendações

*Sempre escreva seu código pensando que quem vai fazer manutenção dele é um psicopata violento que sabe onde você mora*

John Woods, comp.lang.c++

*Quando mais difícil entender e manter seu código, pior o xingamento que você vai receber depois*

Filósofo anônimo

## Exercício

- Escreva um algoritmo para ensinar um esquimó a amarrar o cadarço de seus sapatos
  - Esquimó usa mocassin
  - Esquimó entende português
  - Esquimó sabe o que é cadarço
  - Esquimó não sabe dar laço nem nó

## Conclusão

- Comunicação com computadores
- Programação
- Linguagens de programação
- Tradutores
- Algoritmos

## Perguntas

