



Programação de Computadores I

Introdução

PROFESSORA CINTIA CAETANO

Introdução

▶ Resolução de problemas

- ▶ Encontrar uma maneira de descrever este problema de uma forma clara e precisa.
- ▶ Encontremos uma seqüência de passos que permitam que o problema possa ser resolvido de maneira automática e repetitiva.
- ▶ Esta seqüência de passos é chamada de ALGORITMO.

Algoritmo

- ▶ Receita, procedimento, técnica, rotina.
- ▶ Conjunto finito de regras que fornece uma seqüência de operações para resolver um problema específico.
- ▶ Caminho para a solução de um problema, e em geral, os caminhos que levam a uma solução são muitos.

Programa = Algoritmo + Estruturas de Dados

Algoritmo

- ▶ Um algoritmo opera sobre um conjunto de entradas de modo a gerar uma saída que seja útil (ou agradável) para o usuário.
- ▶ Exemplo:
 - ▶ Fazer um bolo:
 - ▶ Entrada: farinha ovos, fermento, etc.
 - ▶ Saída: Bolo pronto.

Algoritmo

- ▶ **Um algoritmo tem 5 características importantes:**
 - ▶ **Finitude:** Um algoritmo deve sempre terminar após um número finito de passos.
 - ▶ **Definição:** Cada passo de um algoritmo deve ser precisamente definido. As ações devem ser definidas rigorosamente e sem ambiguidades.
 - ▶ **Entradas:** Um algoritmo deve ter zero ou mais entradas, isto é informações que são lidas e fornecidas antes do algoritmo iniciar.
 - ▶ **Saídas:** Um algoritmo deve ter uma ou mais saídas, isto é quantidades que tem uma relação específica com as entradas.
 - ▶ **Efetividade:** Um algoritmo deve ser efetivo. Isto significa que todas as operações devem ser suficientemente básicas de modo que possam ser em princípio executadas com precisão em um tempo finito por um humano usando papel e lápis.

Algoritmo

- ▶ Você sabe construir algoritmos?
- ▶ Claro que sim!!!
- ▶ Se isto não fosse verdade, não conseguiríamos sair de casa pela manhã, ir ao trabalho, decidir qual o melhor caminho para chegar a um lugar, voltar para casa, etc.
- ▶ Para que tudo isto seja feito é necessário uma série de entradas do tipo: a que hora acordar, que hora sair de casa, qual o melhor meio de transporte, etc.

Algoritmo

- ▶ Um exemplo de algoritmo pode ser as instruções que um professor de ginástica passa aos seus alunos em uma academia de ginástica. Por exemplo:
- ▶ Repetir 10 vezes os quatro passos abaixo:
 - ▶ Levantar e abaixar braço direito;
 - ▶ Levantar e abaixar braço esquerdo;
 - ▶ Levantar e abaixar perna esquerda;
 - ▶ Levantar e abaixar perna direita.

Representação de Algoritmos

- ▶ As formas mais comuns de representação de algoritmos são as seguintes:
- ▶ **Linguagem Natural**
 - ▶ Os algoritmos são expressos diretamente em linguagem natural, como nos exemplos anteriores.
- ▶ **Fluxograma Convencional**
 - ▶ Esta é uma representação gráfica que emprega formas geométricas padronizadas para indicar as diversas ações que devem ser executadas e decisões que devem ser tomadas para resolver o problema.
- ▶ **Pseudo-linguagem**
 - ▶ Emprega uma linguagem intermediária entre a linguagem natural e uma linguagem de programação para descrever os algoritmos.

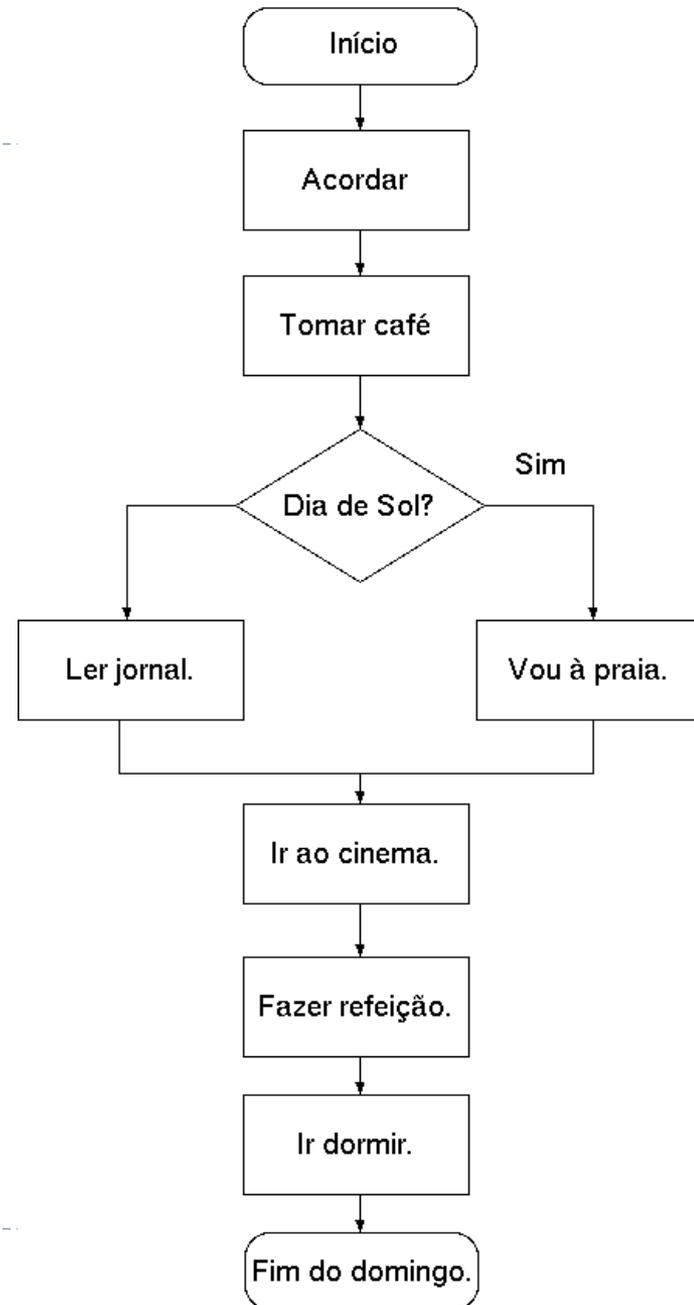
Linguagem Natural

- ▶ Quase todos os algoritmos que apresentamos até este ponto foram escritos em linguagem natural, ou seja, no nosso caso, habitantes do Brasil, o português.

Fluxogramas

- ▶ Esta forma de representação de algoritmos emprega várias formas geométricas para descrever cada uma das possíveis ações durante a execução do algoritmos.

Fluxograma para um domingo



Pseudo-linguagem

- ▶ Também chamado de Pseudo-código.
- ▶ Empregar uma linguagem que esteja o mais próximo possível de uma linguagem de programação de computadores, mas evitando de definir regras de construção gramatical muito rígidas.
- ▶ A idéia é usar as vantagens do emprego da linguagem natural, mas restringindo o escopo da linguagem.

Pseudo-linguagem

- ▶ Calcular a área de uma mesa retangular.
- ▶ Cálculo da área de uma mesa.
 - ▶ Medir a largura da mesa e anotar o resultado.
 - ▶ Medir o comprimento da mesa e anotar o resultado.
 - ▶ Multiplicar o comprimento pela largura e anotar o resultado.
 - ▶ O valor da área da mesa é o resultado anotado no passo anterior.
- ▶ Fim do cálculo da área da mesa.

Pseudo-linguagem

▶ Sintaxe:

algoritmo “NomeDoAlgoritmo”;

 inicio_do_algoritmo

 corpo do algoritmo

 Fim_do_algoritmo

Programa

- ▶ Um programa é um Algoritmo escrito em uma linguagem computacional.

Linguagens de Programação

- ▶ Softwares que permitem o desenvolvimento de programas.
- ▶ Possuem um poder de criação ilimitado, desde jogos, editores de texto, sistemas empresariais até sistemas operacionais.
- ▶ Existem várias linguagens de programação, cada uma com suas características próprias. Por exemplo:
 - ▶ Pascal
 - ▶ C
 - ▶ C++
 - ▶ C#
 - ▶ Delphi
 - ▶ Visual Basic
 - ▶ Java
 - ▶ Etc.

Compilação e Linkedição

- ▶ Quando escrevemos um programa em qualquer linguagem de alto nível (como C, C++, Pascal, Delphi, Java, VisualBasic, etc.), utilizamos um editor de textos para escrever uma série de comandos e códigos que desejamos que o computador execute.
- ▶ Este arquivo contendo estes códigos e comandos em linguagem de alto nível é chamado **código fonte** ou **programa fonte**.

Compilação e Linkedição

- ▶ O computador não é capaz de compreender os comandos contidos neste programa (ou arquivo) fonte, pois a única linguagem que o computador entende é a linguagem de baixo nível, conhecida como **linguagem de máquina**.
- ▶ É necessário realizar um processo de “tradução” que transforma o programa fonte em um programa equivalente escrito em linguagem de máquina.
- ▶ As duas principais maneiras de realizar este processo de tradução são chamadas de **interpretação e compilação**.

Compilação e Linkedição

- ▶ **Interpretação:** o programa interpretador analisa cada linha de seu programa fonte separadamente, verifica se esta linha está correta e, caso esteja, gera uma linha equivalente em linguagem de máquina e realiza a execução. No final, o código em linguagem de máquina resultante da tradução das linhas de seu programa fonte não é guardado.

Compilação e Linkedição

- ▶ **Compilação:** o programa compilador da linguagem analisa todo o programa fonte gerado. Caso nenhum erro seja encontrado, todo o programa fonte será traduzido para uma linguagem de baixo nível e armazenado em um arquivo separado, chamado de arquivo objeto.
- ▶ Um processo adicional chamado de **Linkedição**, transforma este arquivo objeto em um arquivo executável, também capaz de ser compreendido pela máquina. O arquivo executável, como o próprio nome indica, está pronto para ser executado pelo computador.

Vantagens

- ▶ Uma vez gerado o arquivo executável, ele pode ser utilizado inúmeras vezes sem a necessidade da presença do compilador ou qualquer outro utilitário, ao passo que os interpretadores são sempre necessários para a execução de seus programas.
- ▶ Um programa compilado possui uma execução muito mais rápida que um equivalente interpretado, pois a interpretação sempre tem de realizar a tradução de cada linha do seu programa fonte.
- ▶ Compiladores garantem segurança em relação ao seu código-fonte, já que ele não é necessário para a execução de um programa já compilado.