

CURSO DE ENG.SOFTWARE

Disciplina: Algoritmos e Programação

Aula 02

Profº. M.e William P. Santos Júnior

Anápolis – 2021.1

CURSO DE ENG. SOFTWARE

Disciplina: Algoritmos e
Programação I

Introdução à Algoritmos

OBJETIVOS:

- Conhecer Arquitetura de Computadores.
- Modelos de Programação.
- Constantes e Variáveis
- Comandos – Entrada/Saída/Atribuição

CURSO DE ENG. SOFTWARE

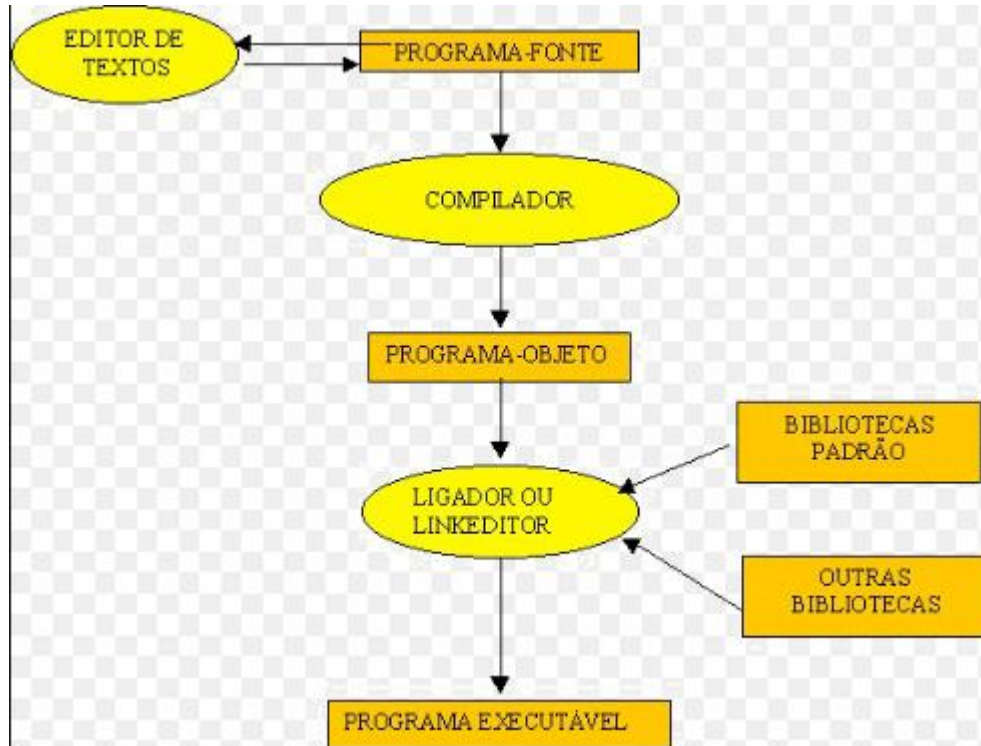
Disciplina: Algoritmos e Programação

INTRODUÇÃO À ALGORITMOS

REFERÊNCIAS:

ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Verenuchi de. Fundamentos da Programação de Computadores: algoritmos, Pascal e C/C++. São Paulo: Pearson Education, 2010.

Conceitos Básicos de Programação:



Conceitos Básicos de Programação:

Código-Fonte: Criado em um editor de textos. Serve como entrada para o compilador.

Código-Objeto: Criado pela conversão do código-fonte em linguagem de máquina. É gerado pelo compilador. Só é criado quando não há erros no código-fonte. (Extensão do código-objeto: .OBJ)

Ligador ou Linkeditor: "junta" o código-objeto com as bibliotecas necessárias para gerar o programa-executável. (Extensões das bibliotecas: .DLL ou .LIB)

Programa Executável: Código que pode ser executado pelo sistema operacional. (Extensão do programa-executável: .EXE)

Tempo de Compilação: Durante o processo de conversão entre código-fonte e código-objeto.

Tempo de Execução: Após a ativação do programa executável.

Lógica do Dia a Dia:

A lógica é a organização do raciocínio
válido

Introdução:

Linguagens Simbólicas(*Assembly*): Linguagens que utilizavam abreviações ou símbolos para transcreverem o código em linguagem de máquina.

Introdução:

Linguagens de Alto Nível: Linguagens atuais, convertem mais rapidamente o código de alto nível para o código de baixo nível, necessitam o uso de *“Compiladores”*.

Introdução:

```
import java.awt.*;
import java.util.*;

interface State
{
    public State get_initial();
    public State get_goal();
    public Stack get_goals(); // Only if problem has multiple goal states
    public State get_copy() throws Exception; // Deep copy
    public boolean equals(State s);
    public void apply(Object operator) throws Exception; // State modifies itself
    public boolean legal_state();
    public Stack possible_operators();
    public void paint(Graphics g); // Only if we want visualization

    // BEGINNING OF A* METHODS (that is, methods needed by A* algorithm)
    // Methods for tracing back along a plan (path through state space)
    public State previousState(); // Returns previous state in plan
    public Object previousOperator(); // Returns operator that changes
    // previous state to this state.

    // Hueristic (h) and Cost(g) functions for this state s: f(s) = g(s) + h(s)
    public double g(); // Current lowest known cost of traversing state space
    // from start state to this state
    public double h(); // Estimated cost of traversing state space
    // from this state to "nearest" goal state
    public double f(); // f() should just return g() + h()

    // If implemented and used, these five methods could save on lookup times
    public boolean on_closed_listQ(); // Returns true iff this state is on closed list
    public boolean on_open_listQ(); // Returns true iff this state is on open list
    public void on_closed_list(boolean yes_no); // Tells this state whether
    // it is now on closed list
    public void on_open_list(boolean yes_no); // Tells this state whether
    // it is now on open list
    public int hash_value(); // Useful if hash table used to store states...
}
```

Compiladores : Programas tradutores, convertem os programas feitos em linguagem de alto nível em linguagem de máquina.

Introdução:

Algoritmos:

1. Sequencia de Passos que visa atingir um objetivo bem definido,
2. Sequencia de Passos que devem ser seguidas para realização de uma tarefas
3. Sequencia finita de instruções ou operações para resolver um problema computacional
4. Regras formais e logicas para solução de problemas

Introdução:

Métodos para Construção de Algoritmos:

- Compreender Completamente o Problema a ser resolvido.
- Definir Dados de Entrada, ou seja, quais dados serão fornecidos para o algoritmo.
- Definir o processamento, ou seja, quais os cálculos serão efetuados.
- Definir os dados de saída, ou seja, quais dados serão gerados depois do processamento.
- Construir o Algoritmo utilizando um tipo específico.
- Testar o Algoritmo realizando simulações.



Introdução:

Algoritmos Narrativos:

Algoritmo 1 – Fazer um Sanduíche

- PASSO 1 - Pegar o pão.
- PASSO 2 - Cortar o pão ao meio.
- PASSO 3 - Pegar a maionese.
- PASSO 4 - Passar a maionese no pão.
- PASSO 5 - Pegar e cortar alface e tomate.
- PASSO 6 - Colocar alface e tomate no pão.
- PASSO 7 - Pegar o hambúrguer.
- PASSO 8 - Fritar o hambúrguer.
- PASSO 9 - Colocar o hambúrguer no pão.

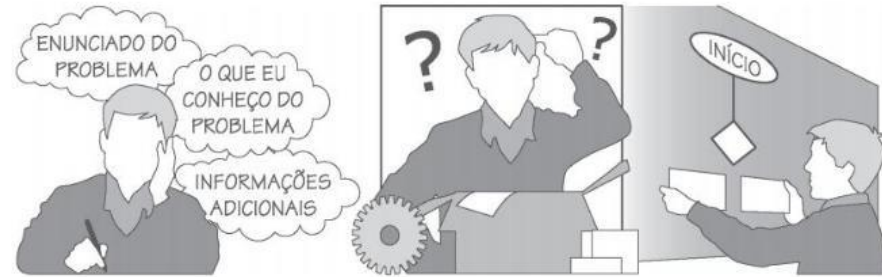


FIGURA 1.3 A tarefa de especificar um algoritmo.

Introdução:

Algoritmos Fluxograma:

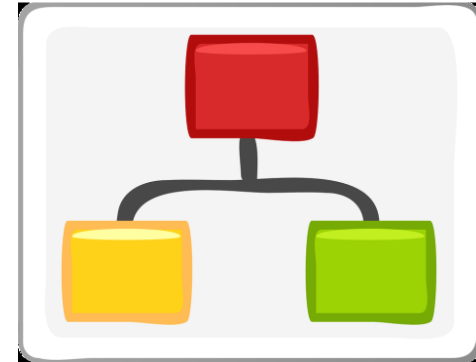
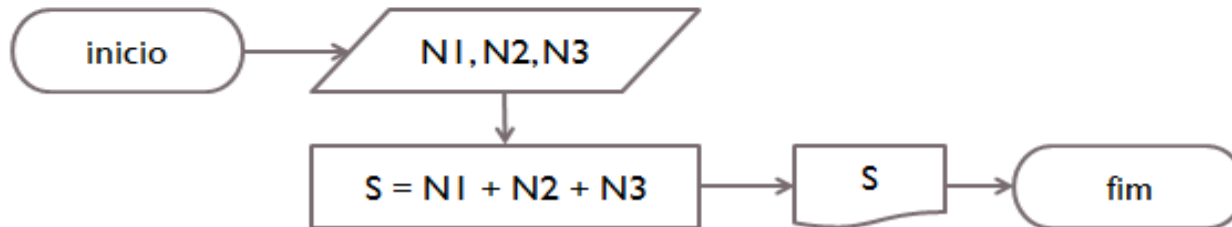
Introdução:

Algoritmo Somar 3 Números:

▶ Descrição Narrativa

- ▶ Passo 1 → Receber do usuário Três números;
- ▶ Passo 2 → Somar os três números;
- ▶ Passo 3 → Mostrar o resultado obtido.

▶ Fluxograma:



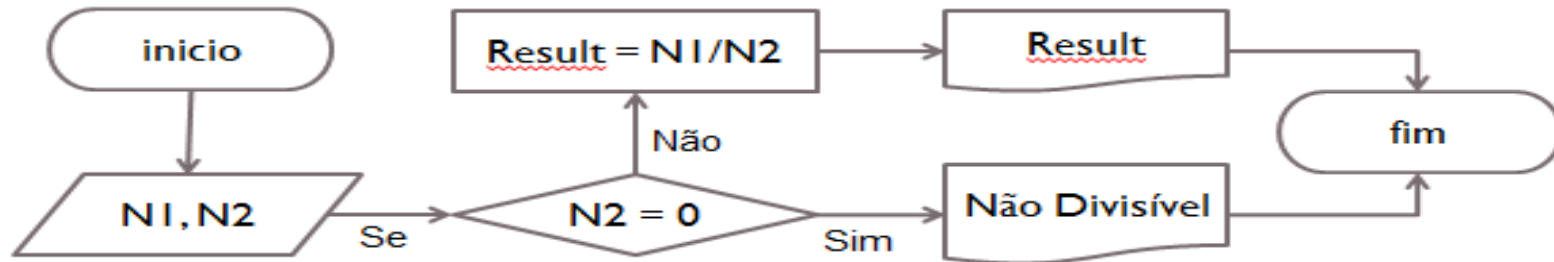
Introdução:

Algoritmo Divisão de 2 Números:




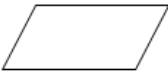


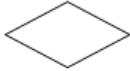
► Descrição Narrativa

- Passo 1 → Receber os dois números;
- Passo 2 → Se o segundo número for igual a zero, não permitir a divisão, caso contrario dividir os números e mostrar o resultado;

► Fluxograma:



Fluxograma:

Símbolo	Função
	Indica o <u>início</u> e o <u>fim</u> do algoritmo.
	Indica o sentido do fluxo de dados. Serve exclusivamente para conectar os símbolos ou blocos existentes.
	Indica cálculos e atribuições de valores.
 	Representa a entrada de dados.
	Representa a saída de dados.
	Representa tomada de decisão, indicando a possibilidade de desvios.

CURSO DE ENG. SOFTWARE

Disciplina: Algoritmos e
Programação

OBJETIVOS:

- Conhecer Arquitetura de Computadores.
- Modelos de Programação.
- Constantes e Variáveis
- Comandos – Entrada/Saída/Atribuição

UniEVANGÉLICA
CENTRO UNIVERSITÁRIO