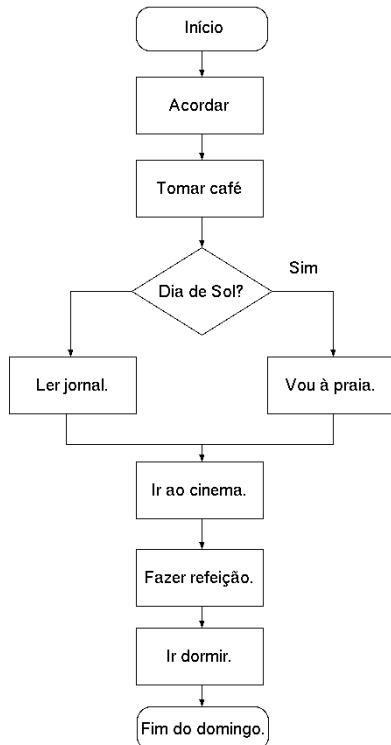


Ciência da Computação

Prof. Dr. Leandro Alves Neves

Fluxograma para um domingo



Aula 03

Sumário

- Linguagens de Programação
- Primeiras Linguagens
- Tipos
- Programa-Fonte
- Tradução

1. Linguagens de Programação

- Pode ser considerada como sendo uma **notação especial**
 - Usada para **especificar procedimentos lógicos**
- As linguagens dão origem aos programas de computador.

2. Primeiras Linguagens

- Uso de linguagem de máquina.
 - Sequências de dígitos binários (0s e 1s).

- Desvantagens:
 - **Erros** em todos os estágios do processo de programação.
 - Endereços de memória: definidos manualmente

 - **Longos programas**
 - Dificuldade no processo de validação e detecção de erros.

3. Tipos de Linguagem

■ Baixo nível

- Linguagem de máquina.
- Forte relação entre as operações implementadas pela linguagem e as operações implementadas pelo hardware.

2001	LDA
2002	5001
2003	LDB
2004	5002
2005	ADD
2006	STO
2007	5003

■ Alto nível

- Próximas das **linguagens utilizadas por humanos**

$$Z = X + Y$$

4. Programa-fonte

- Programa-fonte ou “código-fonte”:
 - Programa escrito em uma linguagem de alto-nível.
 - É “legível”
 - Uso de um programa de edição de textos
 - Por exemplo: Pascal, C, Fortran, Python e outras

5. Processamento de Linguagens

- Um problema básico:
 - Como uma linguagem de nível mais alto pode ser implementada em um computador que usa linguagem de máquina?

- Existem duas alternativas para esta implementação:
 - **Interpretação e**
 - **Tradução (Compilação).**

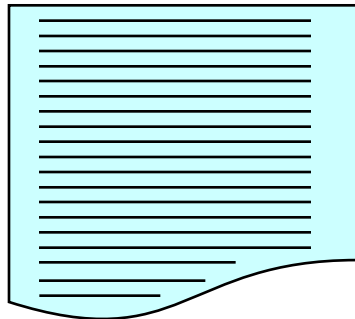
6. Tradução (compilação)

- Chamada genericamente de “**compilação**”.
- Programas escritos em linguagem de alto nível são traduzidos para versões equivalentes em linguagem de máquina antes de serem executados
- Essa tradução é feita em algumas fases:
 - ❑ **Compilação**
 - ❑ **Linkedição**
 - ❑ ***Loader***

6.1 Compiladores

- Produz um código de máquina a partir de um código-fonte.

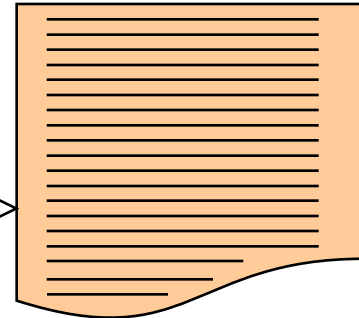
Programa
em Linguagem C



Compilador
(Ling. C)



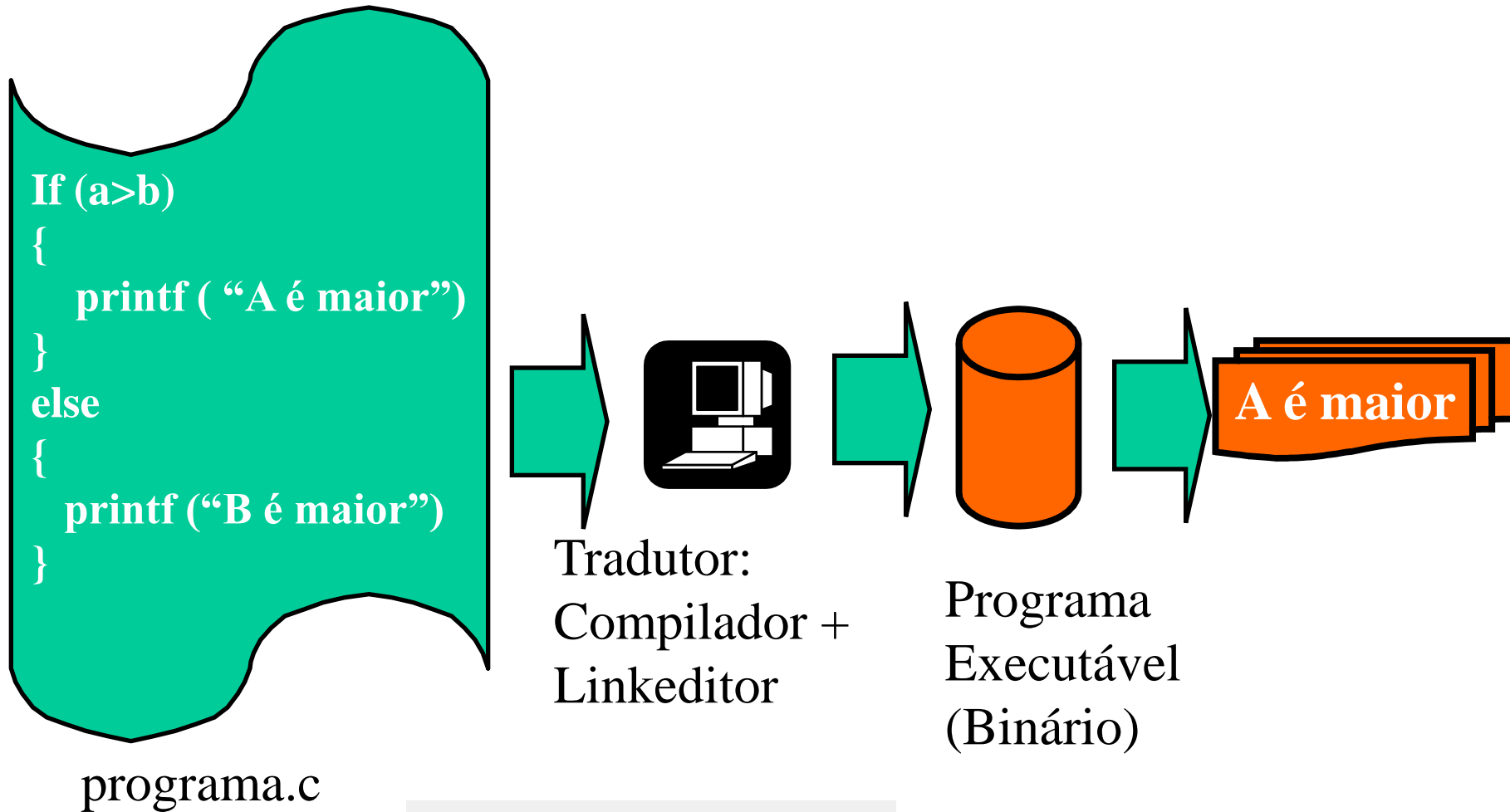
Programa
Máquina



```
#define MAX 1000

int main (void) {
    int v[MAX];
    for (int i = 0; i < MAX; ++i) {
        . . .
    }
    return EXIT_SUCCESS;
}
```

6.1 Compiladores

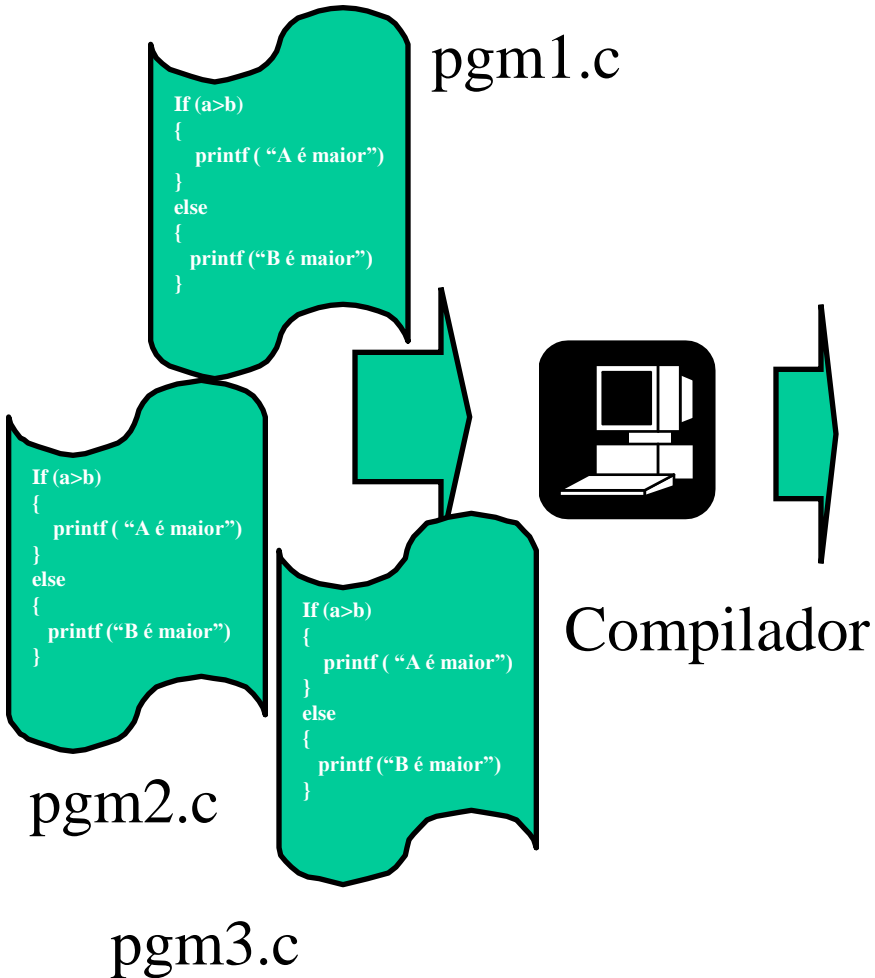


```
~$ gcc programa.c -o res
```

6.1 Compiladores

Fases da Compilação:

1. **Pré-processamento** (tratamento de todas as linhas de código que **começam com #**, por exemplo)



```
#define MAX 1000

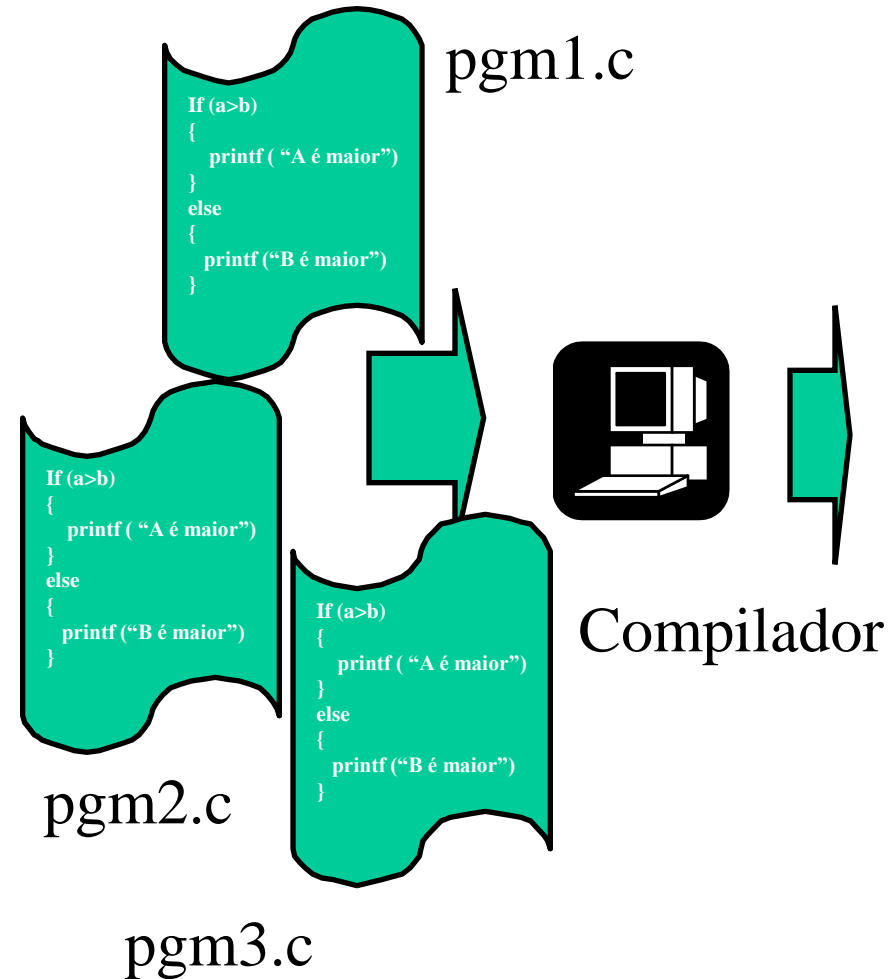
int main (void) {
    int v[MAX];
    for (int i = 0; i < MAX; ++i) {
        . . .
    }
    return EXIT_SUCCESS;
}
```

Transformado em

```
int main (void) {
    int v[1000];
    for (int i = 0; i < 1000; ++i) {
        . . .
    }
    return EXIT_SUCCESS;
}
```

Versão do programa sem diretivas de pré-processamento.

6.1 Compiladores



Fases da Compilação:

2. Compilar arquivos Pré-processados (**Verificação Sintática**)

```
int main (void) {
    int v[1000];
    for (int i = 0; i < 1000; ++i) {
        . . .
    }
    return EXIT_SUCCESS;
}
```

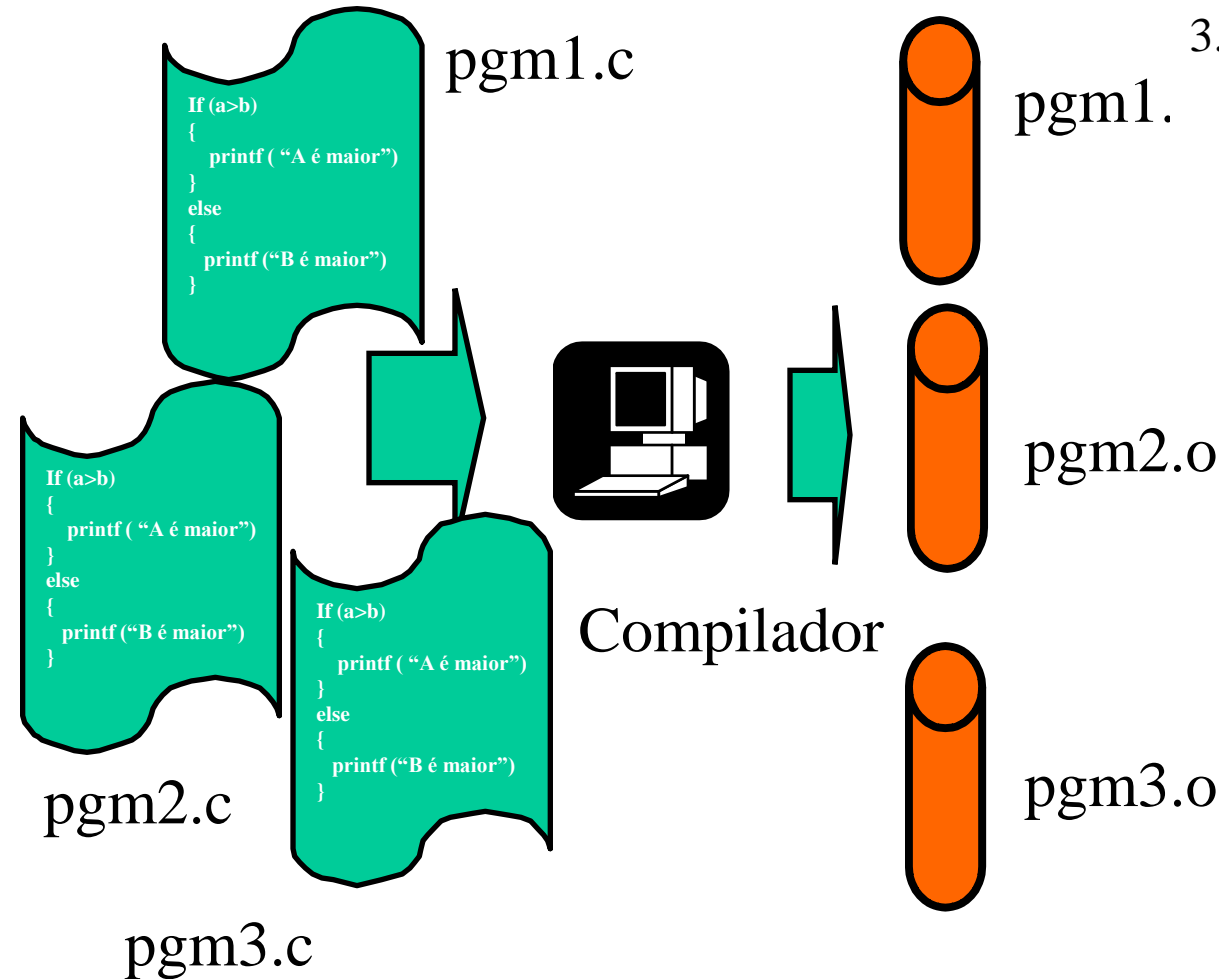
Verificação sintática, tais como parênteses não fechados, falta de ponto-e-vírgula no final da instrução e outros

6.1 Compiladores

Fases da Compilação:

3. Compilação: transforma o código pré-processado em um *programa-objeto*

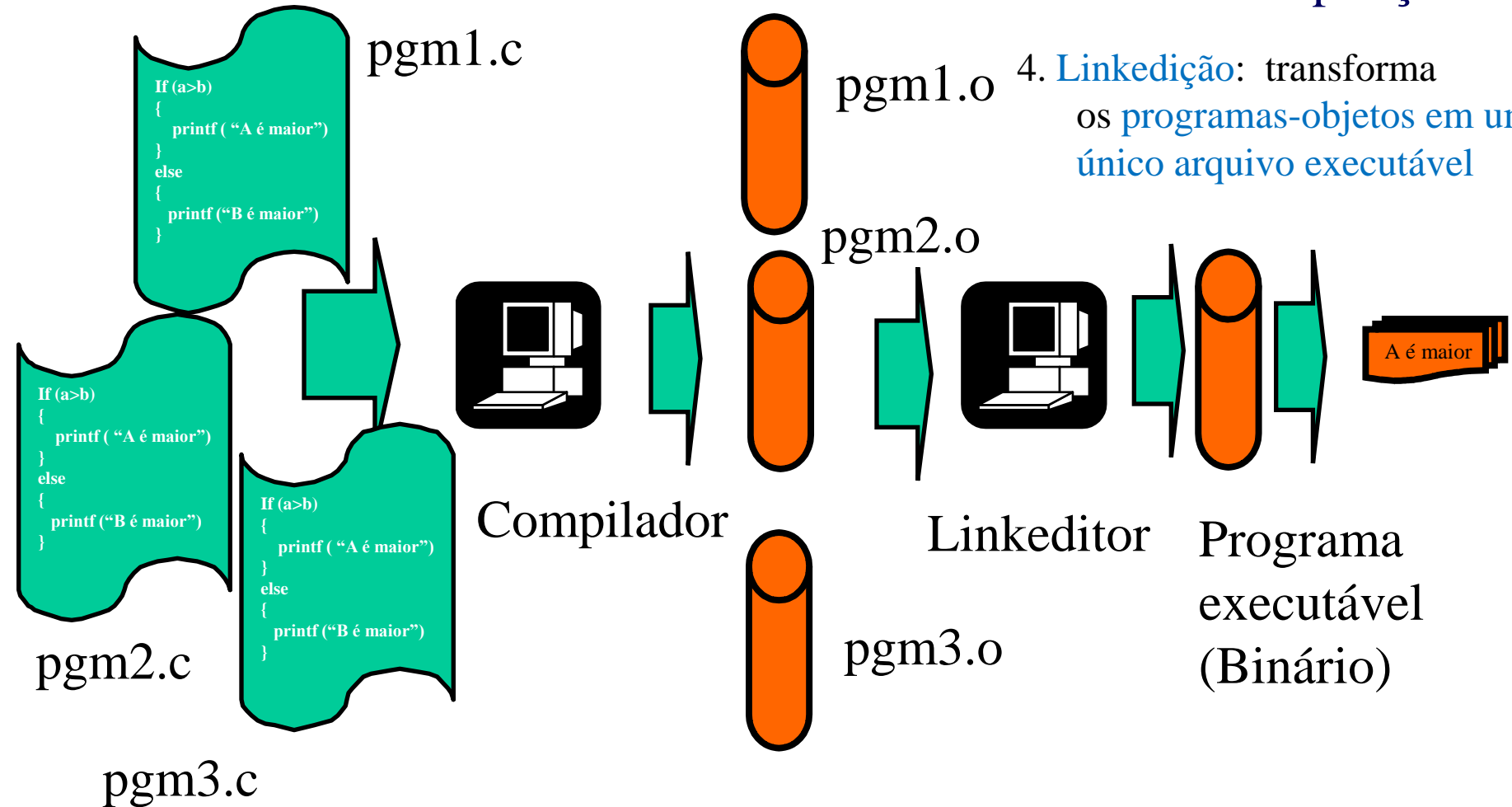
Linguagem de máquina
porém não está pronto para
ser executado



6.1 Compiladores

Fases da Compilação:

4. **Linkedição**: transforma os programas-objetos em um único arquivo executável



- Até aqui vimos o seguinte:
 - A representação da informação
 - Linguagem de máquina, Linguagem de alto nível
 - Tipos de linguagens e suas características
 - Interpretação e Compilação

Bibliografia Complementar

- SCHILDT, H. C Completo e Total, 3ª ed., Pearson 1996. 852p.
 - Parte 1, páginas de 1 a 15
 - Páginas 276 a 278: A Biblioteca C Padrão

