

Linguagem de Programação I - ESP201

Prof^a Ana Carolina Sokolonski

Bacharelado em Sistemas de Informação
Instituto Federal de Ciência e Tecnologia da Bahia
Campus de Feira de Santana

carolsoko@ifba.edu.br

May 27, 2025

Introdução à Linguagem de Programação C

- 1 Breve Histórico
- 2 Vantagens da Linguagem de Programação C
- 3 Palavras Reservadas da Linguagem C
- 4 Geração de Código em C
- 5 Bibliotecas da Linguagem C
- 6 Primeiro Programa na Linguagem C
- 7 Compilando no Linux - GCC
- 8 Usando um Compilador Online
- 9 Referências

Breve Histórico

- Em 1973 Dennis Ritchie reescreveu o sistema operacional UNIX usando a linguagem Linguagem C [Schildt e Mayer 1997]
- Surgem diversas implementações utilizando C, porém os códigos gerados eram incompatíveis
- ANSI (American National Standard Institute), em 1983, padronizou a linguagem C

Aplicações Desenvolvidas em C

- Sistema Operacional: UNIX (todas as variações)
- Planilhas: 1,2,3 e Excel
- Banco de Dados: dBase III, IV e Access (SGBD)
- Aplicações Gráficas: Efeitos Especiais de filmes como Star Trek, Star Wars e Final Fantasy.

Vantagens da Linguagem de Programação C

- Pode-se construir programas organizados e concisos ocupando pouco espaço de memória com alta velocidade de execução (como o Assembly)
- O C foi desenvolvido a partir da necessidade de se escrever programas que utilizassem recursos próprios da linguagem de máquina de uma forma mais simples e portátil que o Assembly
- Portabilidade entre máquinas e sistemas operacionais

Vantagens da Linguagem de Programação C

- Dados compostos em forma estruturada
- Programas Estruturados
- Total interação com o Sistema Operacional
- Alia características de linguagens de alto e baixo nível (liberdade ao programador)
- Código compacto (42 comandos ANSI) e rápido, quando comparado ao código de outras linguagens de programação de complexidade análoga

Palavras Reservadas da Linguagem C

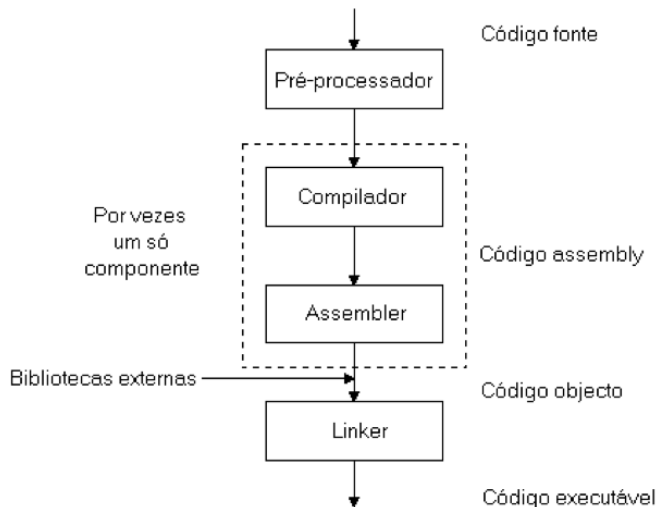
Nomes Reservados (42 comandos)

auto	double	if	static
break	else	int	struct
case	entry	long	switch
char	extern	register	typedef
continue	float	return	union
default	for	sizeof	unsigned
do	goto	short	while

Geração de Código em C

- C é uma linguagem compilada: lê todo o código fonte e gera o código objeto (linguagem de máquina) uma única vez
- Linguagem Interpretada: lê o código fonte, traduz e executa cada vez que o programa é executado.
- O código fonte .C (**source code**) é o programa propriamente dito. Quando compilado, se não apresentar erros é gerado um arquivo .OBJ, após esse processo inicia-se o processo de **Linkedição**, ou seja, junta-se o código sem erros de sintaxe com as bibliotecas de apoio, formando assim o programa executável (.EXE para windows, .DMG para macOS, e assim por diante).

Compilação na Linguagem C



Vantagens da Linguagem de Programação C

A linguagem C é uma linguagem compilada, fortemente tipada, de baixo nível e imperativa

Bibliotecas da Linguagem C

- Conjunto de funções para realizar tarefas específicas
- Biblioteca padrão C - ANSI - funções básicas
- As primeiras linhas do programa indicam as bibliotecas utilizadas:

`#include "minha_biblioteca.h" ou`

`#include <minha_biblioteca.h>`

Primeiro Programa na Linguagem C

```
/* Primeiro Programa em C */  
#include <stdio.h>  
  
void main() {  
    printf("Meu primeiro programa em C\n");  
}
```

Observações Importantes

- Sempre que o código fonte for alterado ele deve ser novamente compilado
- C é “*case sensitive*”
- Deve-se listar antecipadamente todas as variáveis utilizadas no programa (declaração de variáveis);
- Todos comandos devem terminar com ponto e vírgula “;”
- As estruturas de seleção e repetição, além das funções, devem usar “{ }” a não ser que sejam compostas de apenas um comando (isso não vale para funções)

Primeiro Programa na Linguagem C Comentado

```
/* Primeiro Programa em C */ //comentários
#include <stdio.h> /*biblioteca de E/S */

void main() /*função principal - início do programa*/
{
    /*marca início da função*/
    printf("Meu primeiro programa em C\n");
    /*função para escrever na tela*/
} /*marca fim da função*/
```

Desenvolvendo um Código Fonte em C

- Utilizar um editor qualquer: vi ou xedit (Linux)
- No caso do Windows, é aconselhável usar uma IDE (ambiente de desenvolvimento) como o DEV-C ou DEV-C++
- No MacOS pode ser o BBEdit
- Caso não queira instalar uma IDE, use as diversas IDEs online, como o GDB Online
- Digite o programa e salve-o com o nome: **nome.c**

Compilando no Linux - GCC

Para compilar usando o compilador GCC na linha de comando do LINUX: (verifica a sintaxe)

```
>> cc fonte.c <ENTER>
```

Para compilar e gerar executável:

```
>> cc fonte.c -o nomeexec <ENTER>
```

Para executar o arquivo gerado:

```
>> ./nomeexec <ENTER>
```

OBS:

Se o nome do executável não for informado o *default* será *a.out*

Compilando no Linux - GCC

- Para compilar usando o **GCC** na linha de comando do LINUX:

```
>> cc fonte.c < ENTER >
```

- Para compilar e gerar executável:

```
>> cc fonte.c -o nomeexec < ENTER >
```

- Para executar o arquivo gerado:

```
>> ./nomeexec < ENTER >
```

- Se o nome do executável não for informado o *default* é **a.out**

Parâmetros de Compilação no Linux - GCC

- -g: Monta e *linkedita* um código-objeto
- -o: Compila e *linkedita* um arquivo fonte
- -c: Compila, mas não monta ou *linkedita*. O resultado é um código-objeto
- -E: Pré-processa. Todas as diretivas de pré-processamento são executadas, para que você só veja o código simples
- -S: Compila e monta, mas não *linkedita*. Resulta em um código *Assembly*

Parâmetros de Otimização no Linux - GCC

- -O0 (O zero) a -O3 ."0" significa que não há otimização, enquanto o "3" é mais elevado nível de otimização. "1" e "2" estão entre as bordas extremas. Se você só usar -O, sem especificar qualquer número, implicitamente significa -O1.
- -Os : Diz ao GCC para otimizar o tamanho. Basicamente, é semelhante ao -O2 , mas pula algumas etapas que podem aumentar o tamanho do arquivo.

Usando um Compilador Online - GDB Online



The screenshot shows the OnlineGDB website interface. The browser address bar displays "onlinegdb.com". The page features a sidebar on the left with navigation links: "IDE", "My Projects", "Classroom", "Learn Programming", "Programming Questions", "Sign Up", "Login", and "Learn Python with KodeKloud". The main content area is titled "source code" and contains a C program. The program includes a multi-line comment, a preprocessor directive for stdio.h, and a main function that prints "Hello World". The interface includes buttons for "Run", "Debug", "Stop", "Share", "Save", and "Beautify". The language is set to "C".

```
1. /*=====
2.
3. Welcome to GDB Online.
4. GDB online is an online compiler and debugger tool for C, C++, Python, PHP, Ruby,
5. C#, OCaml, VB, Perl, Swift, Prolog, Javascript, Pascal, COBOL, HTML, CSS, JS
6. Code, Compile, Run and Debug online from anywhere in world.
7.
8. =====*/
9. #include <stdio.h>
10.
11. int main()
12. {
13.     printf("Hello World");
14.
15.     return 0;
16. }
17.
```

Referências

Referências



SCHILDT, H.; MAYER, R. *C completo e total*. Pearson Universidades, 1997. ISBN 9788534605953. Disponível em: <<https://books.google.com.br/books?id=Pbl0AAAACAAJ>>.