

Programação de Computadores

Prof.: Maiquel de Brito

31 de julho de 2018

BLU3101 - Introdução à Informática para Automação

Departamento de Engenharias

UFSC Blumenau

Table of contents

1. Introdução
2. Linguagens de Programação
3. Variáveis
4. Operadores aritméticos

Introdução

Escrevendo programas de computador

- Até agora falamos de algoritmos
passos para solucionar um problema
- Ao escrever programas, informamos esses passos ao computador
- Esses passos são informados de forma que o computador
“compreenda”
Linguagem de programação (ex.: Java, Pascal, Python, C, C++)

Escrevendo programas de computador

- Escrever programas de computador é escrever códigos para instruir o computador a fazer alguma coisa
- Essa “coisa” pode ser algo tão simples quanto somar dois números quanto algo complexo como colocar um satélite em órbita

<https://www.youtube.com/watch?v=9pilla0xGCo>

<https://www.youtube.com/watch?v=u0-pfzKbh2k>

Linguagens de Programação

Linguagem de programação

- Método padronizado para expressar instruções para um computador, ou seja, é um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador.
- Permite que um programador especifique precisamente sobre quais dados um computador vai atuar, como estes dados serão armazenados ou transmitidos e quais ações devem ser tomadas sob várias circunstâncias;

Código Fonte

É o algoritmo escrito sob a forma de uma linguagem de programação existente

```
1 #include <stdio.h>
2
3 void main(){
4     printf(" Hello World\n" );
5 }
```


Compilador

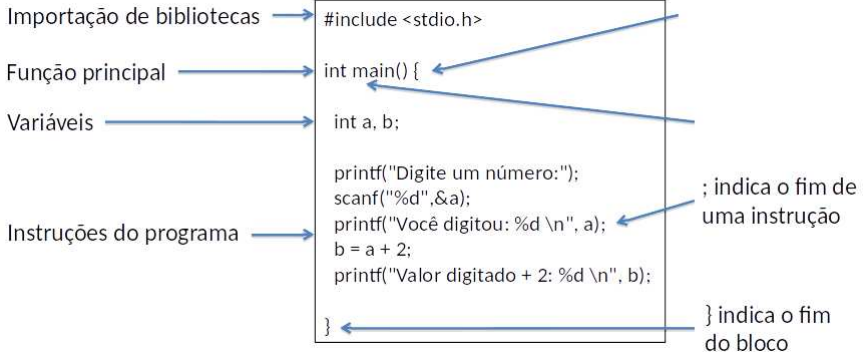
Converte o código fonte escrito em uma linguagem de programação para código executável de máquina (programa)

O processo de compilação

- Entrada: código fonte
- Processamento
 - Análise sintática
 - Análise semântica
- Saída
 - Código fonte correto \rightsquigarrow programa
 - Código fonte incorreto \rightsquigarrow erros de compilação

- Criada por Dennis Ritchie e Brian Kernighan nos laboratórios da Bell na década de 60
- É uma linguagem padronizada pelo ANSI
- Existe uma versão denominada C++ que incorpora características modernas de programação como orientação a objetos etc

Estrutura de programa em linguagem C



Variáveis

Exercício

Criar um programa que pergunte o nome do usuário e em seguida imprima na tela a seguinte mensagem:

Bom dia, <<usuário >>

Variáveis

- Variável é um dado que pode sofrer alterações durante a execução de um algoritmo
- Corresponde a uma posição de memória.
- Variáveis têm um nome e um tipo.

Variáveis

```
1  #include <stdio.h>
2
3  int main(){
4      char nome[80];
5
6      printf("Informe o seu nome: ");
7      scanf("%s",&nome);
8
9      printf("Bom dia , %s\n",nome);
10
11 }
```

Regras para nomes de variáveis na linguagem C

1. começar com letra ou *underscore* (“_”)
 2. os caracteres seguintes podem ser letras, dígitos ou *underscore*
 3. máximo de 32 caracteres
 4. diferente de palavras reservadas, nomes de funções e bibliotecas
- ↪ C é *case sensitive*, i.e. diferencia maiúsculas de minúsculas

Variáveis

Tipo	No.Bits	Formato scanf/printf	Intervalo	
			Início	Fim
char	8	%c	-128	127
unsigned char	8	%c	0	255
signed char	8	%c	-128	127
int	16	%d	-32.768	32.767
unsigned int	16	%d	0	65.535
signed int	16	%i	-32.768	32.767
short int	16	%hi	-32.768	32.767
unsigned short int	16	%hu	0	65.535
signed short int	16	%hi	-32.768	32.767
long int	32	%li	-2.147.483.648	2.147.483.647
signed long int	32	%li	-2.147.483.648	2.147.483.647
unsigned long int	32	%lu	0	4.294.967.295
float	32	%f	3,4E-38	3.4E+38
double	64	%lf	1,7E-308	1,7E+308
long double	80	%LF	3,4E-4932	3,4E+4932

Declaração de variáveis

Formato: <tipo_da_variável> <lista_de_variáveis>;

```
1 char letra ;  
2 int indice , contador ;  
3 float pi ;
```

Leitura de variáveis com scanf

sintaxe: `scanf(string_de_controle, lista_de_variáveis)`

ex.: `scanf("%d",&i) //i é uma variável do tipo int`

`scanf("%c",&c) //c é uma variável do tipo char`

Impressão de variáveis com printf

sintaxe: `printf(string_de_controle, lista_de_variáveis)`

ex.: `printf("O valor a variável é %d",&i) //i é uma variável do tipo int`

`printf("O valor a variável é %d",&c) //c é uma variável do tipo char`

Leitura e impressão de variáveis

```
1 #include <stdio.h>
2
3 int main(){
4     char c;
5     int i;
6     float f;
7
8     printf(" Digite um caractere: ");
9     scanf("%c",&c);
10
11    printf(" Digite um número inteiro: ");
12    scanf("%d",&i);
13
14    printf(" Digite um número decimal: ");
15    scanf("%f",&f);
16
17    printf("Os valores lidos foram: %c, %d, %f", c, i, f);
18    return 0;
19 }
```

O símbolo `=` é o operador de atribuição. Usamos ele para atribuir um valor a uma variável.

Exemplo:

```
a = 10; // armazena o valor 10 na variável a
```

```
x = 'c'; // armazena o caractere 'c' na variável x
```

Operadores aritméticos

Operadores aritméticos

Operador	Operação
+	Soma (números inteiros e decimais)
-	Subtração(números inteiros e decimais)
*	Multiplicação (números inteiros e decimais)
/	Divisão (números inteiros e decimais)
%	Resto de divisão (números inteiros)
++	Incremento (números inteiros e decimais)
--	Decremento (números inteiros e decimais)

Exercício

Escrever um programa que solicite que o usuário digite um número e que, depois, informe o dobro desse número.

```
1 #include <stdio.h>
2
3 int main(){
4     int numero, dobro;
5
6     printf("Informe um número inteiro: ");
7     scanf("%d",&numero);
8
9     dobro = numero*2;
10
11     printf("O dobro de %d é %d\n",numero,dobro);
12 }
```


Estrutura de programa em linguagem C

Primar pela clareza, organização e **identação** do código

```
#include <stdio.h>

int main() {

    int a, b;

    printf("Digite um número:");
    scanf("%d",&a);
    printf("Você digitou: %d \n", a);
    b = a + 2;
    printf("Valor digitado + 2: %d \n", b);

}
```

=

```
#include <stdio.h>

int main() {int a, b;printf("Digite um
número:");scanf("%d",&a);printf("Voc
ê digitou: %d \n", a);b = a +
2;printf("Valor digitado + 2: %d \n",
b);}
```

Referência sobre C

- Programar em C (Wikibook)
http://pt.wikibooks.org/wiki/Programar_em_C
- Livro aberto: Aprendendo a Programar: Programando na Linguagem C
<http://professor.ic.ufal.br/jaime/livros/Aprendendopdf>
- Página atual do compilador Dev-C++
<http://orwelldevcpp.blogspot.com.br/>
- Tutoriais, exemplos e referências <http://www.cprogramming.com/>
- Blockly Games <https://blockly-games.appspot.com/>
- Indentação de código fonte
<http://pt.wikipedia.org/wiki/Indenta%C3%A7%C3%A3o>