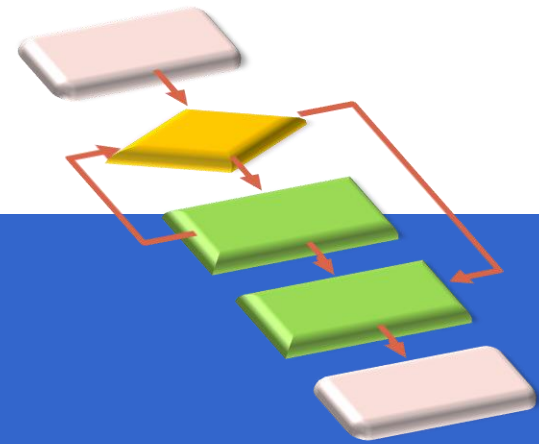




INSTITUTO DE COMPUTAÇÃO



Algoritmos II

CONCEITOS INTRODUTÓRIOS

Comandos básicos e estrutura de um programa

Prof.^a Vanessa de Oliveira Campos

Comando de Atribuição

- Para atribuir valores às variáveis, utiliza-se o símbolo = (sinal de igualdade) para inserir valores ou operações a variáveis.



Comando de Atribuição - Exemplo

```
1  #include <stdio.h>
2  int main ( ) // função principal
3  {
4      // declarações de variáveis
5      int horas;
6      float salario;
7      double dias, desconto;
8      char nome [50] = "Vanessa Campos";
9      char sexo;
10
11     horas = 36;
12     salario = 1200.34;
13     dias = (double) horas / 24;
14     sexo = 'F';
15     printf("Imprimindo valores");
.    // comandos para imprimir o conteúdo de variáveis
.    ...
. }
```

Comando de Saída

- O comando de saída é utilizado para mostrar dados na tela.
- Um comando de saída na linguagem C é o **printf**.

Código	Função escreve
%c	um único caracter
%s	uma série de caracteres
%d	um número inteiro
%u	um inteiro sem sinal
%ld	um inteiro longo
%f	um número em ponto flutuante
%lf	um double

```
int soma;  
float media;  
...  
printf("Olá mundo \n");  
printf("O total é %d", soma);  
printf("A média é %f", media);
```



Comando de Entrada

- O comando de entrada realiza a leitura de dados a partir do teclado.
- Um comando de entrada na linguagem C é **scanf**.
- Parâmetros:
 - Uma *string*, indicando os tipos das variáveis que serão lidas e o formato dessa leitura.
 - Uma lista de variáveis.



Comando de Entrada - Exemplo

```
1  #include <stdio.h>
2  int main ( ) // função principal
3  {
4      int m;
5      float valor;
6      double porcentagem;
7      char genero;
8
9      printf("Digite um valor inteiro: ");
10     scanf(" %d", &m);
11     printf("Digite um valor float: ");
12     scanf(" %f", &valor);
13     printf("Digite um valor double: ");
14     scanf(" %lf", &porcentagem);
15     printf("Digite um caractere: ");
16     scanf(" %c", &genero);
17
18     return 0;
19 }
```

Expressões Aritméticas

Operador		Significado
+	Soma	
-	Subtração	
*	Multiplicação	
/	Divisão. Se os operandos são inteiros, o resultado será um número inteiro. Se os operandos são números fracionários, resultado será fracionário.	
%	Resto da divisão inteira	
++	Incrementa 1 ao valor atual	
--	Decrementa 1 ao valor atual	



Comentários

- Delimitando o texto:
 - entre os símbolos “/*” e “*/”.
- A partir de um ponto na linha:
 - após os caracteres “//”.



Operadores Lógicos

- Na linguagem C, temos os seguintes operadores lógicos:

&& : operador E.

|| : operador OU.

! : operador de NEGAÇÃO.



Operadores Relacionais

Operador	Exemplo	Operador verifica se ...
==	C == D	o conteúdo de C é igual ao conteúdo de D.
!=	C != D	o conteúdo de C é diferente ao conteúdo de D.
<=	C <= D	o conteúdo de C é menor ou igual ao conteúdo de D.
>=	C >= D	o conteúdo de C é maior ou igual ao conteúdo de D.
<	C < D	o conteúdo de C é menor que o conteúdo de D.
>	C > D	o conteúdo de C é maior que o conteúdo de D.



Estrutura de um programa

- Definição do objetivo do programa.
- Inclusão de bibliotecas de funções predefinidas.
- Declarações de constantes.
- Função *main*.



Estrutura de um programa

12

```
1 // comentário onde eh definido o objetivo do programa
2 #include ... // inclusão de bibliotecas de funções predefinidas
3 #include ...
4 #define... // declarações de constante
5 #define...
6 int main ( ) // função principal
7 {
8     // declarações de variáveis
9     <declaracao de variavel_1>
10    <declaracao de variavel_2>
11    ...
12    <declaracao de variavel_n>
13    // comandos
14    Comando_1;
15    Comando_2;
16    ...
17    Comando_n;
18    return 0; // valor devolvido
19 }
```

Exemplo de um programa

13

```
1 //Informa a soma de dois valores lidos
2 #include <stdio.h> // biblioteca de funções de entrada e saída
3 #include <stdlib.h> // biblioteca de funções de controle de vídeo
4 #include <math.h> // biblioteca de funções matemáticas
5 int main ( ) // função principal
6 {
7     float valor1, valor2; // declaração de variáveis
8     float soma;
9     printf ("\nForneca os dois valores a serem somados:");
10    scanf ("%f %f", &valor1, &valor2); // leitura dos valores
11    soma = valor1 + valor2; // calculo da soma
12    printf ("\nSoma dos dois valores: %8.2f\n", soma); // saída
13    system("pause"); //segura a tela de execução
14    return 0; // valor devolvido
15 }
```

