

TIPO DE DADOS

ALGORITMO	LINGUAGEM C-ANSI
inteiro	int
real	float double
caracter	char

Tipo	Tamanho	Intervalo	Uso
char	1 byte	-128 a 127	número muito pequeno e caracter ASCII
int	2 bytes	-32768 a 32767	contador, controle de laço
float	4 bytes	3.4e-38 a 3.4e38	real (precisão de 7 dígitos)
double	8 bytes	1.7e-308 a 1.7e308	científico (precisão de 15 dígitos)

DECLARAÇÃO DE VARIÁVEIS

ALGORITMO	LINGUAGEM C-ANSI
<u>nome do tipo</u> <i>identificadores</i> ;	<i>tipo lista de identificadores</i>
EX: <u>inteiro</u> codigo, CPF; <u>real</u> preco, nota; <u>caracter</u> nome	EX: float nota; int código; double CPF; char nome[31];
<u>inteiro</u> notas[30]; <u>inteiro</u> baixa[4][6];	int altura[30]; float peso[30][5];

DEFINIÇÃO DE CONSTANTES

ALGORITMO	LINGUAGEM C-ANSI
<u>defina</u> <i>lista de identificadores valor</i>	#define PI 3.14159
EX: <u>defina</u> PI 3,14159	#define ON 1 #define OFF 0 #define ENDERECO 0x378 void main() {

COMENTÁRIOS

ALGORITMO	LINGUAGEM C-ANSI
delimitado por chaves { }	delimitado por /* */
EX: <u>declare</u> nota <u>numérico</u> {nota do aluno}	EX: float nota; /* nota do aluno */

EXPRESSÕES ARITMÉTICAS

ALGORITMO	LINGUAGEM C-ANSI
adição: +	+
subtração: -	-
multiplicação: *	*
divisão: /	/
radiciação: $\sqrt{\quad}$	sqrt(valor)
potenciação: n^y	pow(n,y)
resto da divisão: resto(div.,divisor)	div % divisor
quoc. da divisão: quoc(div.,divisor)	div / divisor
ATRIBUIÇÃO:	ATRIBUIÇÃO:
EX: nota ← 10	EX: nota = 10

EXPRESSÕES LÓGICAS

ALGORITMO	LINGUAGEM C-ANSI
igual a: =	==
diferente de: <>	!=
maior que: >	>
menor que: <	<
maior ou igual a: >=	>=
menor ou igual a: <=	<=
conjunção: <u>e</u>	&&
disjunção: <u>ou</u>	
negação: <u>não</u>	!

EXPRESSÕES LITERAIS

ALGORITMO	LINGUAGEM C-ANSI
concatenação:	strcat(A,B)
comparação: =	strcmp(A,B)
atribuição: ←	strcpy(A,B)
EX: A contém "bola" B contém "preta" A B = "bolapreta" A = B resposta será falsa B ← A resposta será B = "bola"	EX: A contém "bola" B contém "preta" strcat(A,B) retorna A = "bolapreta" strcmp(A,B) resposta será valor <> 0 strcpy(B,A) resposta será B = "bola"

COMANDOS DE ENTRADA DE DADOS

ALGORITMO	LINGUAGEM C-ANSI
<u>leia</u> lista de identificadores	scanf("código de formatação",&ident)
EX: <u>leia</u> nota	EX: scanf("%d", ¬a)

COMANDOS DE SAÍDA DE DADOS

ALGORITMO	LINGUAGEM C-ANSI
<u>escreva</u> ident. e/ou constantes	printf("código de formatação/constantes",ident.)
EX: <u>escreva</u> "A nota do aluno foi:", nota	EX: printf("A nota do aluno foi %d", nota)

CÓDIGO DE FORMATAÇÃO	FORMATOS
%c	caracter (um único)
%d	decimal (int)
%i	inteiro
%e	ponto flutuante - notação científica
%f	ponto flutuante (float)
%s	string
%p	ponteiro

COMANDOS DE ENTRADA DE DADOS

ALGORITMO	LINGUAGEM C-ANSI
<u>leia</u> lista de identificadores	scanf("código de formatação",&identificador)
EX: <u>leia</u> nota	EX: scanf("%d", ¬a)

CONTROLE / CARACTER	SEQUENCIA DE ESCAPE
nulo (null)	\0
campainha (bell)	\a
retrocesso (backspace)	\b
tabulacao horizontal	\t

nova linha (new line)	\n
tabulacao vertical	\v
alimentacao de folha (form feed)	\f
retorno de carro (carriage return)	\r
aspas (")	\"
apostrofo (')	\'
interrogacao (?)	\?
barra invertida (\)	\\

ESTRUTURA SEQUENCIAL

ALGORITMO	LINGUAGEM C-ANSI
<pre> algoritmo {exemplo 1} declare a, b, c <u>numérico</u> <u>leia</u> a, b c ← (a + b) * b <u>escreva</u> a, b, c <u>fim algoritmo</u> </pre>	<pre> #include <stdio.h> #include <conio.h> main () { int a, b, c; printf("Digite o valor de A"); scanf("%d",&a); printf("Digite o valor de B"); scanf("%d",&b); c = (a+b)*b. printf("A=%d, B = %d, C = %d",a,b,c); } </pre>

ESTRUTURA CONDICIONAL

ALGORITMO	LINGUAGEM C-ANSI
<pre> algoritmo {exemplo 2} declare a, b c, menor <u>numérico</u> <u>leia</u> a, b, c <u>se</u> (a > b) <u>e</u> (a < c) <u>então</u> menor ← a <u>senão se</u> (b < c) <u>então</u> menor ← b <u>senão</u> menor ← c <u>fim se</u> <u>escreva</u> "o menor valor é:",menor <u>fim algoritmo</u> </pre>	<pre> #include <stdio.h> #include <conio.h> main () { int a, b, c, menor; printf("Digite o valor de A"); scanf("%d",&a); printf("Digite o valor de B"); scanf("%d",&b); printf("Digite o valor de C"); scanf("%d",&c); if ((a < b) && (a < c)) { menor = a; } else { if (b < c) { menor = b; } else { menor = c; } } printf("O menor valor eh: %d",menor); } </pre>

ESTRUTURA DE REPETIÇÃO - 1ª FORMA: Interrupção no Início

ALGORITMO	LINGUAGEM C-ANSI
<pre> EX.: algoritmo {exemplo 3} declare i <u>numérico</u> i ← 1 </pre>	<pre> EX.: LAÇO FOR #include <conio.h> #include <stdio.h> main() </pre>

<pre> repita se (i > 100) então interrompa fim se escreva "O número eh:", i i ← i + 1 fim repita fim algoritmo </pre>	<pre> { int i; clrscr(); for (i = 1; i <= 100; i++) { printf("O número eh: %d", i); } } </pre> <p>EX.: LAÇO WHILE</p> <pre> #include <conio.h> #include <stdio.h> main() { int i = 1; clrscr(); while (i <= 100) { printf("O número eh: %d", i); i++; } } </pre>
--	--

ESTRUTURA DE REPETIÇÃO - 2ª FORMA: Interrupção no Interior

ALGORITMO	LINGUAGEM C-ANSI
<pre> algoritmo {exemplo 4} declare nome, sobrenome, nomec litera repita leia nome se nome="fim" então interrompa fim se leia sobrenome nomec ← nomec nome nomec ← nomec " " nomec ← nomec sobrenome escreva "O nome completo da pessoa eh:", nomec fim repita fim algoritmo </pre>	<pre> #include <conio.h> #include <stdio.h> #include <string.h> main() { char nome[11], sobrenome[11], nomec[22] clrscr(); printf("Primeiro nome: "); scanf("%s", &nome); while (strcmp(nome, "fim") != 0) { printf("Sobrenome: "); scanf("%s", &sobrenome); strcpy(nomec, " "); strcat(nomec, nome); strcat(nomec, " "); strcat(nomec, sobrenome); printf("O nome completo da pessoa eh: %s", nomec); printf("Primeiro nome: "); scanf("%s", &nome); } } </pre>

ESTRUTURA DE REPETIÇÃO - 3ª FORMA: Interrupção no Final

ALGORITMO	LINGUAGEM C-ANSI
<pre> algoritmo {exemplo 5} declare i numérico i ← 1 repita escreva "O número eh:", i i ← i + 1 se (i >= 100) então interrompa fim se fim repita fim algoritmo </pre>	<pre> #include <conio.h> #include <stdio.h> main() { int i = 0; clrscr(); do { printf("O número eh: %d", i); i++; } while (i <= 100) } </pre>