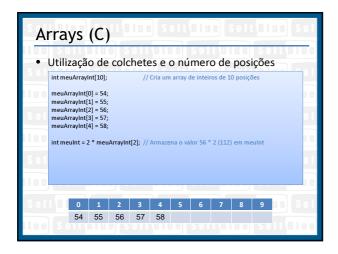
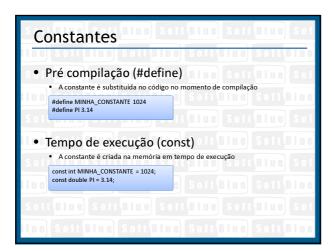


Introdução ao Objective C Também conhecido como ObjC Baseada na linguagem C Criada por Brad Cox e Tom Love em 1980 Licenciada em 1988 pela NeXT Principal linguagem de desenvolvimento para o Mac Linguagem dinâmica: maior velocidade Permite criação de classes em tempo real Utilização do Xcode neste módulo do curso

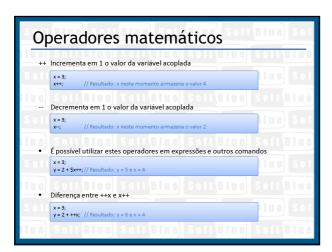
char	TO BILLE SUIT BILLE SUIT
char meuChar; char meuChar = "texto"; char meuChar[10]; char meuChar[10] = "textotexto"; char meuChar[] = "texto";	// char de uma única posição apenas declarado // char de uma única posição já inicializado // char de 10 posições apenas declarado // char de 10 posições já inicializado // char cujo número de posições será cakulado pelo // compliador, no caso, 6 (5 letras + final de string)
nto i ma le a sa lo la	HA CASEDINA CASEDINA
int meulnt; int meulnt = 50;	// variável do tipo int apenas declarada // variável do tipo int já inicializada
loat	
float meuFloat; float meuFloat = 50.25;	// variável do tipo float apenas declarada // variável do tipo float já inicializada
louble	
double meuDouble; double meuDouble = 50.25252;	// variável do tipo double apenas declarada // variável do tipo double já inicializada
oool males of tRIII	

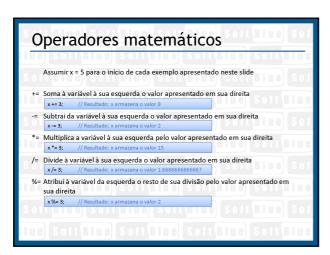




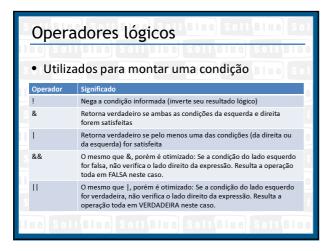
Operadores e comandos						
llue Soit Blue Soit Blue Soit Blue Soit Blue						
Operadores matemáticos						
Operadores condicionais						
Operadores lógicos						
Comandos de controle						
 Comandos de repetição 						
Soft Blue Soft Blue Soft Blue Soft Blue Sof						
liue Soit Biue Soit Biue Soit Biue Soit Biue						
Saft Blue Saft Blue Saft Blue Saft Blue Saf						
lue soit Blue soit Blue soit Blue soit Blue						

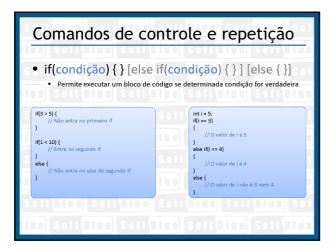
Para os exemn	los apresentados, assuma:			
double x;	// Variável "x" declarada como do tipo de dado double			
Atribui o valor	da sua direita à variável a sua esquerda			
x = 3;	// Resultado: x armazena o valor 3			
Soma dois valo	ores numéricos			
x = 3 + 5;	// Resultado: x armazena o valor 8			
Subtrai dois va	lores numéricos			
x = 3 - 5;	// Resultado: x armazena o valor -2			
Multiplica dois valores numéricos				
x = 2 * 6;	// Resultado: x armazena o valor 12			
Divide dois valores numéricos				
x = 18 / 3;	// Resultado: x armazena o valor 6			
Obtém o resto	da divisão entre dois valores numéricos			
x = 19 % 3;	// Resultado: x armazena o valor 1			



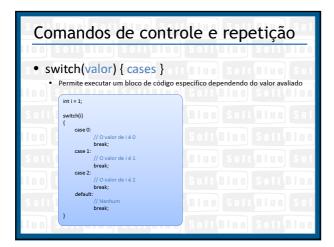


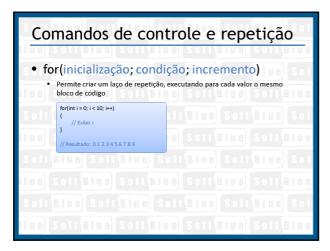
Operadores condicionais • Utilizados para avaliar uma condição Operador Significado == Compara se dois valores tem o mesmo valor, mesmo em tipos distintos === Compara se dois valores são idênticos != Compara se dois valores são diferentes !== Compara se dois valores são diferentes !== Compara se dois valores são diferentes inclusive no tipo de dado ! Inverte o resultado (negação) < Compara se o valor da direita é menor que o da esquerda > Compara se o valor da direita é maior que o da esquerda <= Compara se o valor da esquerda é menor ou igual que o da direita >= Compara se o valor da esquerda é maior ou igual que o da direita





Comandos de controle e repetição							
Short if (if reduzido) Forma reduzida de escrever o comando if Deve ser utilizado somente em casos onde o bloco de código for curtíssimo							
s • condição ? bloco_true : bloco_false; HIHO S o I							
int x = 5; int y; if(x % 2 == 0) {	Soft Lue Soft	int x = 5; int y; y = x % 2 == 0 ? 60 : 61; // Se x for par, atribuirá 60, caso contrário 61					
Hue zont Blue zon	TIB I u e	SOTTBILLE SOTTBILLE					





Comandos de controle e repetição							
 while(condição) Permite executar várias vezes um bloco de código enquanto sua condição for verdadeira 							
Soft Sing S	int i = 0; while(i < 5) { // Exibir i !++;	Soft Blue Soft Blue Soft Blue) f				
Soft S	} // Resultado: 0 1 2 3 4	SoftBlue SoftBlue SoftBlue SoftBlue) f				
Soft S	Blue Soft Blue Soft <mark>Blue</mark> Soft	SoftBlue SoftBlue SoftBlue SoftBlue	o f				

