



Linguagem C

Revisão: controle condicional



Baseado em slides do Prof. Bruno Travençolo

Comando if

- ▶ Em linguagem C, o comando if é utilizado quando for necessário escolher entre dois caminhos, ou quando se deseja executar um comando sujeito ao resultado de um teste.
- ▶ A forma geral de um comando **if** é:
if (expressão)
 instrução
- ▶ A expressão, na condição, será avaliada:
 - ▶ Se ela for zero (falsa), a instrução não será executada;
 - ▶ Se a condição for diferente de zero (verdadeira) a instrução será executada.



Exemplo if

```
int main()
{
    int num;
    printf("Digite um numero: ");
    scanf("%d",&num);
    if (num==10){
        printf("O numero eh igual a 10.\n");
    }
    return 0;
}
```



Comando if

- ▶ Pode-se usar chaves { } para delimitar o bloco de instruções que pertence ao if

```
if (num > 10) {  
    printf ("\n\n O numero eh maior que 10");  
}
```

- ▶ As chaves **devem** ser usadas no caso de mais de uma instrução:

```
if (nota >= 60) {  
    printf ("A nota é maior ou igual a 60 \n") ;  
    printf ("O aluno está aprovado!") ;  
}
```

- ▶ As chaves podem ser ignoradas se a instrução for única.

```
if (num > 10)  
    printf ("\n\n O numero e maior que 10") ;
```



Exercício

- ▶ Dada a idade de um paciente, monte a expressão if que verifica se ele precisará se vacinar. O paciente deverá se vacinar se sua idade for maior ou igual a 2 e menor do que 60 anos.



Exercício

```
#include <stdio.h>
#include <stdlib.h>

int main (){
    int num;
    printf ("Digite a idade: ");
    scanf ("%d",&num);

    if ((num >= 2) && (num < 60)){
        printf ("O paciente deve ser vacinado \n");
    }

    return 0;
}
```



Comando else

- ▶ O comando else pode ser entendido como sendo um complemento do comando if.
 - ▶ Se o **if** diz o que fazer quando a condição é verdadeira, o **else** trata da condição falsa.
- ▶ O comando if-else tem a seguinte forma geral:
if (expressão)
 instrução 1
else
 instrução2



Exemplo if-else

```
int main()
{
    int num;
    printf("Digite um numero: ");
    scanf("%d",&num);
    if (num==10){
        printf("O numero eh igual a 10.\n");
    } else {
        printf("O numero eh diferente de 10.\n");
    }
    return 0;
}
```



Comando if-else

- ▶ Como no caso do comando if, as chaves podem ser ignoradas se a instrução contida no **else** for única.

```
if (num==10){  
    printf("O numero eh igual a 10.\n");  
} else // else sem usar chaves  
    printf("O numero eh diferente de 10.\n");
```

▶ ou

```
if (num==10){  
    printf("O numero eh igual a 10.\n");  
} else { // else com chaves  
    printf("O numero eh diferente de 10.\n");  
}
```



Exercício: complementando o anterior

- ▶ Dada a idade de um paciente, monte a expressão if que verifica se ele precisará se vacinar. O paciente deverá se vacinar se sua idade for maior ou igual a 2 e menor do que 60 anos. Nesse caso, deve ser impresso um aviso para que ele se vacine. Caso ele não deva vacinar, deve ser impresso um aviso também (para não vacinar).



Exercício

```
#include <stdio.h>
#include <stdlib.h>

int main (){
    int num;
    printf ("Digite a idade: ");
    scanf ("%d",&num);

    if ((num >= 2) && (num < 60)){
        printf ("O paciente deve ser vacinado \n");
    } else {
        printf ("O paciente nao deve ser vacinado \n");
    }

    return 0;
}
```



Aninhamento de if

- ▶ O **if** aninhado é simplesmente um **if** dentro da declaração de um outro **if** externo.
 - ▶ A estrutura if-else-if é apenas uma extensão da estrutura if-else.
- ▶ O único cuidado que devemos ter é o de saber exatamente a qual **if** um determinado **else** está ligado.



Aninhamento de if

```
if(condição 1) {  
    seqüência de comandos 1;  
} else if(condição 2) {  
    seqüência de comandos 2;  
}  
...  
else if(condição_n) {  
    seqüência de comandos n;  
} else {  
    seqüência de comandos default;  
}
```



Aninhamento de if

- ▶ O programa começa a testar as condições começando pela 1 e continua a testar até que ele ache uma expressão cujo resultado dê diferente de zero (verdadeiro). Neste caso ele
 - ▶ executa a seqüência de comandos correspondente.
 - ▶ Só uma seqüência de comandos será executada, ou seja, só será executada a seqüência de comandos equivalente à condição que der diferente de zero (a que for encontrada primeiro).
 - ▶ A última seqüência de comandos (default) é a que será executada no caso de todas as condições darem zero (falso) e é opcional.



Exemplo aninhamento

```
01  #include <stdio.h>
02  #include <stdlib.h>
03  int main(){
04      int num;
05      printf("Digite um numero: ");
06      scanf("%d", &num);
07      if(num == 10){
08          printf("O numero e igual a 10.\n");
09      } else{
10          if(num > 10)
11              printf("O numero e maior que 10.\n");
12          else
13              printf("O numero e menor que 10.\n");
14      }
15      system("pause");
16      return 0;
17  }
```



Exercício

- ▶ Dada o valor da nota de um aluno, monte o conjunto de if's e else's que verifica se ele foi aprovado (acima ou igual a 60), reprovado (abaixo de 50) ou reprovado com RM (entre 50 e 60).



Exercício

```
#include <stdio.h>
#include <stdlib.h>
```

```
int main ()
{
    int nota;

    printf ("Digite a nota: ");
    scanf ("%d",&nota);

    if (nota >= 60){
        printf ("Aluno aprovado");
    } else {
        if (nota < 50){
            printf ("Aluno reprovado");
        } else {
            printf ("Aluno reprovado com RM");
        }
    }

    return 0;
}
```

Exercício

- ▶ Construir a seqüência de if-else para escrever o nome do dígito lido
 - ▶ '0' -> “zero”;
 - ▶ '1' -> “um”;
 - ▶ etc.



Exercício

```
char ch;  
scanf("%c",&ch);  
if (ch == '0') printf("Zero");  
else if (ch=='1') printf("Um");  
else if (ch=='2') printf("Dois");  
else if ...  
else if (ch=='9') printf("Nove");  
else printf("Nao era um digito!");  
...
```



O comando switch

- ▶ O comando switch é próprio para se testar uma variável em relação a diversos valores pré-estabelecidos.
 - ▶ Parecido com if-else-if, porém não aceita expressões, **apenas constantes**.
 - ▶ O switch testa a variável e executa a declaração cujo *case* corresponda ao valor atual da variável.



O comando switch

```
switch (expressão) {  
  case valor1:  
    comandos1;  
    break;  
  case valor2:  
    comandos2;  
    break;  
  ....  
  case valor k:  
    comandos k;  
    break;  
  default:  
    comandos default;  
}
```



O comando switch

- ▶ O comando switch

- ▶ Avalia o valor de expression com os valores associados às cláusulas **case** em seqüência;
 - ▶ Quando o valor associado a uma cláusula é igual ao valor de expression os respectivos comandos são executados até encontrar um **break**.
- ▶ A declaração **default** é opcional e será executada apenas se a expressão expression que está sendo testada não for igual a nenhuma das constantes presentes nos **case**.



O comando switch

```
01  #include <stdio.h>
02  #include <stdlib.h>
03  int main(){
04      char ch;
05      printf("Digite um simbolo de pontuacao: ");
06      ch = getchar();
07      switch( ch ) {
08          case \': printf("Ponto.\n" ); break;
09          case \',: printf("Virgula.\n" ); break;
10          case \': printf("Dois pontos.\n" ); break;
11          case \';: printf("Ponto e virgula.\n"); break;
12          default : printf("Nao eh pontuacao.\n" );
13      }
14      system("pause");
15      return 0;
16  }
```



O Operador ?

- ▶ Também conhecido como operador ternário
- ▶ A expressão condicional “? :” é uma simplificação do if-else utilizada tipicamente para atribuições condicionais
- ▶ Exemplo: Dados dois números x e y, retorne o maior na variável z:
 - ▶ Usando if-else
 - ▶ Usando o operador ternário



Exercício

	Usando if-else	Usando operador ternário
01	#include <stdio.h>	#include <stdio.h>
02	#include <stdlib.h>	#include <stdlib.h>
03	int main(){	int main(){
04	int x,y,z;	int x,y,z;
05	printf("Digite x:");	printf("Digite x:");
06	scanf("%d",&x);	scanf("%d",&x);
07	printf("Digite y:");	printf("Digite y:");
08	scanf("%d",&y);	scanf("%d",&y);
09	if (x > y)	z = x > y ? x : y;
10	z = x;	printf("Maior = %d\n",z);
11	else	system("pause");
12	z = y;	return 0;
13	printf("Maior = %d\n",z);	}
14	system("pause");	
15	return 0;	
16	}	

