

Estruturas de Controle de Fluxo

1. Instrução condicional (continuação)

Exercício 2:

Construa um programa em C que receba através da entrada padrão um valor inteiro pertencente ao intervalo $[0,255]$ e a posição do bit que se deseja verificar se está ativo, o programa deve indicar na saída padrão o resultado desta verificação.

```

#include <stdio.h>
int main ()
{
    unsigned char desloca, byte, aux=1;
    printf("\nDigite um numero pertencente ao intervalo [0,255]: ");
    scanf("%hhd", &byte);
    printf("\nDigite a posicao do bit a testar entre 0 e 7: ");
    scanf("%hhd", &desloca);
    aux = aux << desloca;
    byte = byte & aux;
byte = byte >> desloca;
    if (byte = 1)
        printf("\n 0 bit testado esta ativo.\n");
    else
        printf("\n 0 bit testado nao esta ativo.\n");
}

```

5 = 0 0 0 0 0 1 0 1
&
0 0 0 0 0 1 0 0 = 4

```
#include <stdio.h>
int main ()
{
    unsigned char desloca, byte, aux=1;
    printf("\nDigite um numero pertencente ao intervalo [0,255]: ");
    scanf("%hhd", &byte);
    printf("\nDigite a posicao do bit a testar entre 0 e 7: ");
    scanf("%hhd", &desloca);
    aux = aux << desloca;
    byte = byte & aux;
    if (byte)
        printf("\n 0 bit testado esta ativo.\n");
    else
        printf("\n 0 bit testado nao esta ativo.\n");
}
```

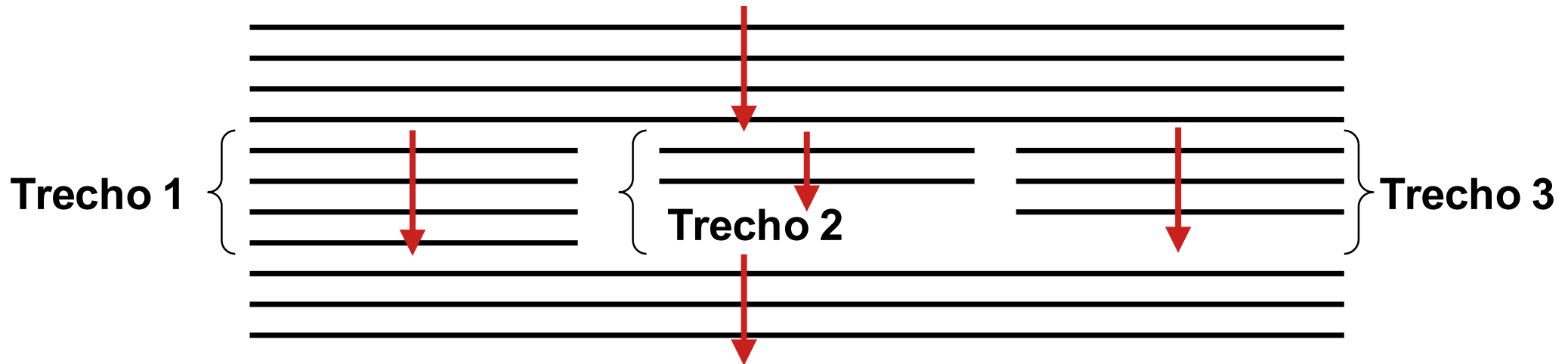


Estruturas de Controle de Fluxo

Comando de Seleção Múltipla (switch)

Estruturas de Controle de Fluxo

2. Comando de seleção múltipla



Estruturas de Controle de Fluxo

2. Comando de seleção múltipla

```
switch (<identificador_variável_escalar>
{
    case <constante1> : <instrução1> ou <instruções1>
                        break;
    case <constante2> : <instrução2> ou <instruções2>
                        break;
    . . .
    case <constanten> : <instruçãon> ou <instruçõesn>
                        break;
    default:           <instruçãon+1> ou <instruçõesn+1>
}

```

Estruturas de Controle de Fluxo

2. Comando de seleção múltipla

Exemplo:

O programa a seguir ao receber um número inteiro pertencente ao intervalo [1, 12] retorna o nome do mês correspondente ao valor fornecido.

```
#include <stdio.h>
int main()
{
    int mes;
    printf("\nEntre com um valor inteiro de 1 a 12 : ");
    scanf("%d",&mes);
    switch (mes)
    {
        case 1: printf ("0 %dº mes do calendario eh Janeiro", mes);
                break;
        case 2: printf ("0 %dº mes do calendario eh Fevereiro", mes);
                break;
        case 3: printf ("0 %dº mes do calendario eh Marco", mes);
                break;
        case 4: printf ("0 %dº mes do calendario eh Abril ", mes);
                break;
        case 5: printf ("0 %dº mes do calendario eh Maio", mes);
                break;
```



```
case 6: printf ("0 %dº mes do calendario eh Junho", mes);  
        break;  
case 7: printf ("0 %dº mes do calendario eh Julho", mes);  
        break;  
case 8: printf ("0 %dº mes do calendario eh Agosto", mes);  
        break;  
case 9: printf ("0 %dº mes do calendario eh Setembro", mes);  
        break;  
case 10: printf ("0 %dº mes do calendario eh Outubro", mes);  
         break;  
case 11: printf ("0 %dº mes do calendario eh Novembro", mes);  
         break;  
case 12: printf ("0 %dº mes do calendario eh Dezembro", mes);  
         break;  
default: printf ("Nao existe um mes que corresponda a %s",  
                "este valor.");  
}  
}
```

Exercício 3

```
#include <stdio.h>
int main()
{
    int a,b,c;
    printf("\nEntre com o valor de a : ");
    scanf("%d",&a);
    switch (a)
    {
        case 1:
            b=2;
            break;
        case 2:
            {
                c=3;
                b=a*c;
            }
            break;
        case 3 :
            c=a;
        default:
            b=500;
    }
    printf("\nvalor de a = %d    valor de b = %d    valor de c = %d\n",a,b,c);
}
```

Determine:

Saídas para a==1

valor de a = 1 valor de b = 2 valor de c = ?

Saídas para a==2

valor de a = 2 valor de b = 6 valor de c = 3

Saídas para a==3

valor de a = 3 valor de b = 500 valor de c = 3

Saídas para a==11

valor de a = 11 valor de b = 500 valor de c = ?



Estruturas de Controle de Fluxo

2. Comando de seleção múltipla

Exercício 4: Construa um programa, na linguagem C, que receba dois inteiros e utilizando a estrutura de controle de fluxo *switch*, ofereça ao usuário um menu com as opções de soma, subtração, multiplicação e divisão. Após o usuário fazer sua opção o programa deve efetuar a operação aritmética selecionada tendo como operandos os valores fornecidos, apresentando o resultado na saída padrão.

```
#include <stdlib.h>
#include <stdio.h>
int main()
{
    int opcao,n1,n2,resultado;
    printf("\nEntre com o primeiro valor inteiro: ");
    scanf("%d",&n1);
    printf("\nEntre com o segundo valor inteiro: ");
    scanf("%d",&n2);
    printf("Digite:\n1 - para somar os inteiros\n2 - para subtrair os inteiros\n");
    printf("3 - para multiplicar os inteiros\n4 - para dividir os inteiros\n0opcao: ");
    scanf("%d",&opcao);
```

```
switch (opcao)
{
    case 1 : resultado=n1+n2;
            break;
    case 2 : resultado=n1-n2;
            break;
    case 3 : resultado=n1*n2;
            break;
    case 4 : resultado=n1/n2;
            break;
    default: printf("Opcao invalida!");
            exit(1);
}
printf("\n0 resultado da operacao foi %d\n",resultado);
}
```



```
switch (opcao)
{
    case 1 : resultado=n1+n2;
            break;
    case 2 : resultado=n1-n2;
            break;
    case 3 : resultado=n1*n2;
            break;
    case 4 : if (n2)
                resultado=n1/n2;
            else
            {
                printf("Divisao por zero!");
                exit(2);
            }
            break;
    default: printf("Opcao invalida!");
            exit(1);
}
printf("\n0 resultado da operacao foi %d\n", resultado);
}
```

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

int main() {
    int n1,n2,resultado;
    char opcao;
    printf("\nEntre com o primeiro valor inteiro: ");
    scanf("%d",&n1);
    printf("\nEntre com o segundo valor inteiro: ");
    scanf("%d",&n2);
    printf("Digite:\n"+"+" para somar os inteiros\n\n"- para subtrair os inteiros\n\n"*" para multiplicar os inteiros\n");
    printf("\n/" para dividir os inteiros\n\n0opcao: ");
    scanf("%c",&opcao);
```

```
switch (opcao) {
    case '+': resultado=n1+n2;
              break;
    case '-': resultado=n1-n2;
              break;
    case '*': resultado=n1*n2;
              break;
    case '/': if (n2)
              resultado=n1/n2;
              else {
                printf("Divisao por zero!");
                exit(2);
              }
              break;
    default : printf("Opcao invalida!");
              exit(1);
}
printf("\n0 resultado da operacao foi %d\n",resultado);
}
```



```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    float f;
    char c;
    printf ("Entre com um real: ");
    scanf ("%f",&f);
    printf ("Entre com um caractere: ");
    fflush (stdin); /*para limpar o buffer*/
    scanf ("%c",&c);
    printf ("\n|%c|%d\n",c,c);
}
```

**Não deve ser utilizado!
Por quê?**



Solução adequada!

```
#include <stdio.h>
int main()
{
    float f;
    char c;
    printf ("Entre com um real: ");
    scanf ("%f",&f);
    printf ("Entre com um caractere: ");
    setbuf (stdin, NULL); /*para limpar o buffer*/
    scanf ("%c",&c);
    printf("\n|%c|%d\n",c,c);
}
```

```
#include <stdlib.h>
#include <stdio.h>
int main()
{
    int n1,n2,resultado;
    char opcao;
    printf("\nEntre com o primeiro valor inteiro: ");
    scanf("%d",&n1);
    printf("\nEntre com o segundo valor inteiro: ");
    scanf("%d",&n2);
    printf("Digite:\n"+"+" - para somar os inteiros\n\n"-\" para subtrair os inteiros\n");
    printf("\n"*\" para multiplicar os inteiros\n\n"/\" para dividir os inteiros\n\nOpcao: ");
    setbuf (stdin, NULL);
    scanf ("%c",&opcao);
    ...
}
```