

**Exercício 1**

Editar, compilar e executar o seguinte programa em C

```
void main(){
    int num;
    printf("\nDigite um número inteiro: ");
    scanf("%d", &num);
    if (num%2==0){
        printf("\nO Número digitado é par!\n");
    }else{
        printf("\nO Número digitado é impar!\n");
    }
}
```

Em seguida, alterar o programa para verificar se o número é par ou ímpar e se for par verificar se a sua metade é par ou ímpar.

O programa deve imprimir mensagens relativas às ocorrências:

```
O número digitado é par
O número digitado é ímpar
A metade do número par é par
A metade do número par é ímpar.
```

**Exercício 2**

Dados dia, mês (número do mês) e ano de uma data, determinar e imprimir a quantidade de dias desde o início do ano até essa data especificada.

Dica: a quantidade de dias dos meses varia e o número de dias do mês de fevereiro depende do ano ser bissexto ou não.

Exemplo: Até a data “13 de março de 2005” transcorreu, durante o ano em curso, a quantidade de 72 dias.

**Exercício 3**

Codificar um programa fonte em C para verificar se um número de CPF é válido ou inválido (verificar se os dígitos de controle são corretos), imprimindo mensagem informativa.

O número de CPF é formado por 11 algarismos, sendo os dois últimos, dígitos de controle.

Indicando os algarismos do CPF por  $d_i$ , vamos representá-lo por  $d_0d_1d_2d_3d_4d_5d_6d_7d_8d_9d_{10}$

A regra para calcular os dígitos de controle  $d_9$  e  $d_{10}$  é:

- Calcular a Soma\_1 =  $10d_0 + 9d_1 + 8d_2 + 7d_3 + 6d_4 + 5d_5 + 4d_6 + 3d_7 + 2d_8$
- Calcular o resto da divisão de Soma\_1 por 11.
- Se o resto < 2 então  $d_9 = 0$ , em caso contrário  $d_9 = 11 - \text{resto}$ .
- Calcular a Soma\_2 =  $11d_0 + 10d_1 + 9d_2 + 8d_3 + 7d_4 + 6d_5 + 5d_6 + 4d_7 + 3d_8$
- Calcular a Soma\_3 = Soma\_2 +  $2d_9$
- Calcular o resto da divisão de Soma\_3 por 11
- Se o resto for menor do que 2 definir  $d_{10} = 0$ , em caso contrário,  $d_{10} = 11 - \text{resto}$

Utilize as instruções descritas abaixo para realizar a “entrada” do número de CPF e a “separação” de seus dígitos.

```
char cpf[13];
int d[11];
int k;

printf("digite o cpf na forma XXXXXXXXX-XX: ");
scanf("%s", &cpf);

for(k = 0; k < 11; k++){
    if(k < 9) d[k]=cpf[k]-48;
    else d[k]=cpf[k+1]-48;
}
```

Depois de realizada a entrada do CPF e a separação de seus dígitos (instruções acima), calcule as somas descritas anteriormente e os dígitos de controle correspondentes, compare os dígitos calculados com os que foram lidos e imprima a mensagem correspondente.

## Tarefa Extra 06 (entregar no início da próxima aula)

Considere o seguinte problema:

*Dado um número inteiro  $N$  imprimir os seus divisores próprios ou a mensagem 'PRIMO'.*

Apresentar:

- A descrição do algoritmo que resolve o problema
- O desenho da tela para a entrada e a saída de dados utilizado na codificação.
- O programa fonte em C que implementa o algoritmo
- Resultado de um ou mais testes do programa
- O trabalho deve ser apresentado em um editor de texto.

## Exercícios complementares

Para cada exercício, o aluno deve descrever um algoritmo e a codificação correspondente.

---

1. Dada uma relação de números inteiros deseja-se verificar a sua divisibilidade em relação aos números 2, 3 e 5. Determinar e imprimir a quantidade de números da relação que são divisíveis apenas por um dos três, quantos são divisíveis por dois quaisquer, quantos são divisíveis pelos três e quantos não são divisíveis por nenhum deles.

---

2. Dados os coeficientes de uma equação de segundo grau,

$$ax^2 + bx + c = 0, a \neq 0$$

determinar as raízes reais, se existir. Imprimir os valores e mensagens se as raízes são reais distintas, dupla ou se não existe raiz real.

---