

Aula Prática 3

Prazo final: 25/03/2018

22 de março de 2018

Instruções gerais

Use um `scanf` para cada entrada, ou seja, se você tem que ler nove dígitos do teclado, use a função `scanf` nove vezes, uma para cada valor a ser lido.

Não utilize a função `system("pause")` nos arquivos a serem submetidos.

Não utilize mensagens para a entrada dos dados (ex.: "Entre com o numero 1: "), utilize a função `printf` apenas para imprimir as mensagens descritas nas atividades.

1 Menor, maior e divisível (menormaior.c)

Faça um programa que leia cinco números inteiros e identifique:

1. O maior valor informado,
2. O menor valor informado,
3. Quantos números são divisíveis por 3.

A entrada deve ser cinco números inteiros, fornecidos pelo teclado. Use a função `scanf` 5 vezes, uma para cada número a ser lido.

A saída deve ser as seguintes mensagens de texto utilizadas na função `printf`:

"Maior: %d\n"

"Menor: %d\n"

"Divisiveis por 3: %d\n"

2 Infracao (infracao.c)

Um motorista que ultrapassa a velocidade máxima permitida estará sujeito a uma infração média, grave ou gravíssima. Faça um programa que receba dois valores: a velocidade máxima de uma via e a velocidade registrada por um radar. A seguir o programa deve imprimir na tela se o motorista será multado ou não, se sim, deve apresentar também a gravidade da infração. Considere que as multas são definidas conforme a tabela abaixo:

Excesso de velocidade sobre a máxima permitida	Natureza da infração
Até 20%	Média
De 20% até 50%	Grave
Acima de 50%	Gravíssima

Instruções para a correta avaliação no Prático:

A entrada deve ser dois números reais fornecidos pelo teclado, portanto use a função `scanf` duas vezes. O primeiro número corresponde a velocidade máxima de uma via e o segundo a velocidade registrada por um radar. A saída deve ser uma das seguintes mensagens de texto utilizando a função `printf`: "Velocidade invalida!\n"

"Nenhuma infracao cometida\n"

"Infracao media\n"

"Infracao grave\n"

"Infracao gravissima\n"

3 Aposentadoria (aposentadoria.c)

A redação atual do parágrafo 1º do artigo 40 da Constituição Federal estabelece que um servidor público poderá se aposentar voluntariamente desde que tenha cumprido um tempo mínimo de exercício no serviço público e atenda uma das seguintes condições:

a) 60 anos de idade e 35 de contribuição, se homem, e 55 anos de idade e 30 de contribuição, se mulher; ou,

b) 65 anos de idade, se homem, e 60 anos de idade, se mulher.

Faça um programa que receba a idade, o tempo de contribuição e o sexo de um servidor público e imprima na tela se esse servidor pode se aposentar ou não. Utilize o caractere "M" para representar o sexo masculino e "F" para representar o sexo feminino. Considere que a idade não pode ultrapassar 70 anos, pois neste caso a aposentadoria é compulsória.

Instruções para a correta avaliação no Prático:

A entrada deve ser nesta ordem:

- Um caractere para representar o sexo do servidor;
- Um inteiro para representar a idade do servidor;
- Um inteiro para representar o tempo de contribuição.

Faça 3 `scanf`, um para cada entrada. A saída deve ser uma das seguintes mensagens de texto utilizando a função `printf`:

```
"Idade invalida!\n"
"Sexo invalido!\n"
"Tempo de contribuicao invalido!\n"
"O servidor nao pode aposentar\n"
"O servidor pode aposentar\n"
"Aposentadoria compulsoria\n"
```

4 Algarismos Romanos (romanos.c)

Utilize o comando `switch` para fazer um programa que leia um número inteiro entre 1 e 9 e imprima este valor em algarismo romano. A numeração romana pode ser consultada [clikando aqui](#).

Instruções para a correta avaliação no Prático:

A entrada deve ser um número inteiro. A saída deve ser uma das seguintes mensagens de texto utilizando a função `printf`:

```
"I\n"
"II\n"
"III\n"
"IV\n"
"V\n"
"VI\n"
"VII\n"
"VIII\n"
"IX\n"
"Numero nao suportado\n"
```

5 CPF (cpf.c)

Dígito verificador ou algoritmo de controle é um mecanismo de autenticação utilizado para verificar a validade e a autenticidade de um valor numérico, evitando dessa forma fraudes ou erros de transmissão ou digitação.

Sua tarefa é implementar um programa que calcule o dígito verificador de um CPF, dado um número de CPF sem o dígito verificador, seu trabalho consiste em calcular o dígito verificador, e imprimir na tela o CPF com o dígito verificador. Você pode testar sua implementação com seu CPF ou de amigos e parentes.

O CPF é composto de 9 dígitos com mais dois dígitos verificadores: $x_1x_2x_3 . x_4x_5x_6 . x_7x_8x_9 - y_1y_2$, onde x_1, \dots, x_9 são os números do CPF, e y_1, y_2 são os dígitos verificadores. Abaixo estão descritos os passos de como se calcular os dígitos verificadores de um CPF, usando como exemplo o número 333.938.741

1. Para obter o primeiro dígito verificador, calcule a soma dos produtos de cada dígito verificador com seu peso, o peso W_n de cada número começa com 10, e diminui de 1 da esquerda para a direita.

$$\sum_{n=1}^9 W_n * x_n$$

A tabela abaixo sumariza essa soma:

Posicao	1	2	3	4	5	6	7	8	9	10 (dv1*)	11 (dv2**)
CPF (x_n)	3	3	3	9	3	8	7	4	1	-	-
W_n	10	9	8	7	6	5	4	3	2	-	-
$W_n * x_n$	30	27	24	63	18	40	28	12	2	-	-
$\sum_{n=1}^i n * x_n$	30	57	81	144	162	202	230	242	244	-	-

*: Dígito verificador 1, como ainda não temos esse valor, ele não é incluído na conta.

** : Não precisamos do dígito verificador 2 para calcular os dígitos verificadores do CPF.

2. Dividimos o número obtido no passo anterior por 11, caso o resto dessa divisão seja 0 ou 1, o primeiro dígito verificador é igual a **0**. Se este não for o caso, o primeiro dígito verificador é $11 - \text{o resto da divisão}$.

Continuando nosso exemplo, $244 \bmod 11 = 2$, caímos no segundo caso, e nosso primeiro dígito verificador é: $11 - 2 = \mathbf{9}$.

3. O passo pra calcular o segundo dígito verificador é semelhante ao primeiro passo, exceto que agora incluímos o primeiro dígito verificador, e os pesos W_n começam a partir de 11.

$$\sum_{n=1}^{10} W_n * x_n$$

Posicao	1	2	3	4	5	6	7	8	9	10 (dv1)	11 (dv2)
CPF (x_n)	3	3	3	9	3	8	7	4	1	9	-
W_n	11	10	9	8	7	6	5	4	3	2	-
$W_n * x_n$	33	30	27	72	21	48	35	16	3	18	-
$\sum_{n=1}^i n * x_n$	33	63	90	162	183	231	266	282	285	303	-

4. A regra para se obter o segundo dígito verificador é a mesma do passo 2: $303 \bmod 11 = 6$, portanto, caímos no segundo caso, e o segundo dígito verificador é: $11 - 6 = 5$.

Assim sendo, o CPF com os dígitos verificadores é 333.938.741-**95**.

Instruções para a correta avaliação no Prático:

A entrada deve ser nove números inteiros, fornecidos pelo teclado, que correspondem ao CPF sem os dígitos verificadores. **Faça um scanf para cada um dos números.** A saída deve ser a seguinte mensagem de texto utilizando a função printf:

"**Dígitos verificadores: %d%d\n**"