

Lista de Exercícios 2 (2023/1)

Este trabalho consiste em resolver a lista de exercícios das páginas a seguir, em C++.

Para a resolução e entrega devem ser seguidas as seguintes regras:

- criar uma pasta com o nome do aluno no formato *camelHump* (por exemplo, para João Pedro da Silva, usar `JoaoPedroDaSilva`);
- dentro dessa pasta criar programas em C++ para resolver cada um dos exercícios, salvando o código-fonte em um arquivo com o nome `Exercicio` seguido do número do exercício com três dígitos (por exemplo, `Exercicio001.cpp`, `Exercicio002.cpp`, ..., `Exercicio100.cpp`);
- no início de cada arquivo em C++, incluir um comentário informando o nome do arquivo, o nome do autor, a finalidade do programa e a versão (ou data) de criação (ou atualização);
- se houver dados para serem lidos, eles devem ser lidos na mesma ordem em que eles são citados no enunciado, escolhendo os tipos numéricos adequadamente;
- escrever os resultados sempre na mesma ordem em que eles são citados no enunciado, escolhendo os tipos numéricos adequadamente (números reais devem ser apresentados sempre com 4 casas decimais, salvo se indicado de outra forma);
- na versão final, tomar o cuidado de não imprimir nada diferente da saída esperada (não devem aparecer, por exemplo, mensagens pedindo que o usuário forneça ou digite determinado valor no terminal);
- a entrega deverá ser feita no dia e horário informado pelo professor em sala de aula e/ou definida na opção de entrega da plataforma moodle da PUCRS;
- cada aluno deverá submeter os códigos-fontes compactados no formato ZIP, usando o mesmo nome da pasta (por exemplo, para João Pedro da Silva, o arquivo compactado deverá chamar-se `JoaoPedroDaSilva.zip`).

7. Faça um programa em C++ que receba dez idades, pesos e alturas, calcule e mostre:

- a média das idades das dez pessoas;
- a quantidade de pessoas com peso superior a 90 kg e altura inferior a 1,50 metro; e
- a porcentagem de pessoas com idade entre 10 e 30 anos entre as pessoas que medem mais de 1,90 m.

Utilize apenas os fluxos de entrada e saída da linguagem C++ (`cin` e `cout`, respectivamente). **NÃO** utilize nem `scanf`, nem `printf`.

Adaptado de: Ascencio e Campos (2012, p. 147)

Exemplos:

Entrada			Saída		
62	66.0	2.1	52.8000	2	25.0000
59	113.2	1.3			
47	94.2	1.7			
21	85.2	2.0			
56	54.2	2.1			
70	95.5	1.3			
69	113.9	2.0			
58	66.5	1.0			
24	53.5	0.9			
62	57.5	0.8			
23	95.4	1.0	48.3000	6	0.0000
58	105.4	1.3			
97	99.5	2.0			
48	98.7	1.4			
10	98.0	1.0			
88	97.9	1.0			
31	119.8	2.1			
34	102.1	1.4			
43	90.4	1.9			
51	93.2	2.0			
24	110.7	1.91	23.2000	0	100.0000
21	117.7	1.92			
24	24.9	2.0			
24	113.4	2.1			
27	49.1	2.05			
21	44.5	2.0			
16	86.8	1.93			
29	79.9	1.92			
25	114.3	2.0			
21	57.1	2.2			
23	94.1	2.0	34.8000	1	75.0000
8	79.8	1.3			
29	76.0	2.1			
44	88.6	1.6			
24	80.2	2.0			
20	39.3	1.7			
71	55.2	1.4			
93	53.6	1.5			
5	117.7	0.9			
31	79.0	2.1			
99	62.1	1.6	51.4000	3	33.3333
83	118.5	1.4			
44	107.9	0.9			
74	60.1	1.3			
25	99.6	2.2			
35	50.3	1.5			
41	97.9	1.4			
97	113.2	2.1			
11	86.3	1.3			
5	102.9	2.2			
96	105.2	1.0	43.8000	1	100.0000
55	86.6	1.3			
39	107.0	1.5			
25	43.0	2.0			
28	63.3	1.2			
32	94.7	1.5			
20	72.3	2.2			
35	61.8	1.7			
60	79.2	1.5			
48	68.3	1.1			
34	26.6	1.7	38.1000	2	50.0000
23	92.6	1.1			
88	85.2	0.7			
52	59.9	1.0			
17	53.0	1.0			
32	98.3	1.4			
48	33.1	0.7			
13	89.6	2.1			
19	61.7	1.7			
55	34.4	2.2			
67	57.5	1.6	51.2000	1	100.0000
27	69.2	1.91			
18	85.8	1.3			
99	44.3	1.7			
51	41.0	0.9			
59	106.9	1.7			
37	63.6	1.6			
17	114.9	1.4			
91	67.6	0.7			
46	116.8	1.8			
93	35.7	0.9	44.2000	1	66.6667
17	98.9	1.6			
49	36.5	1.96			
41	89.3	1.8			
20	70.7	2.1			
78	77.2	1.8			
94	62.2	0.8			
16	79.7	1.95			
23	95.1	0.7			
11	22.9	1.1			
36	79.3	0.8	55.2000	1	50.0000
24	39.1	1.0			
95	101.7	1.8			
67	102.5	2.0			
63	38.7	1.8			
51	82.0	1.9			
87	48.6	0.9			
7	93.1	1.0			
28	40.2	2.2			
94	49.4	1.4			

8. Faça um programa em C++ que leia um vetor com nove números inteiros, calcule e mostre os números primos e suas respectivas posições.
Utilize apenas os fluxos de entrada e saída da linguagem C++ (cin e cout, respectivamente). **NÃO** utilize nem scanf, nem printf.

Adaptado de: Ascencio e Campos (2012, p. 159)

Exemplos:

Entrada									Saída	
1	2	3	4	5	6	7	8	9	2	1
									3	2
									5	4
									7	6
10	11	12	13	14	15	16	17	18	11	1
									13	3
									17	7
20	21	22	23	24	25	26	27	28	23	3
30	31	32	33	34	35	36	37	38	31	1
									37	7
40	41	42	43	44	45	46	47	48	41	1
									43	3
									47	7
50	51	52	53	54	55	56	57	58	53	3
60	61	62	63	64	65	66	67	68	61	1
									67	7
70	71	72	73	74	75	76	77	78	71	1
									73	3
80	81	82	83	84	85	86	87	88	83	3
90	91	92	93	94	95	96	98	99	100	

9. Elabore um programa em C++ que: leia uma matriz 6×4 de inteiros; recalcule a matriz digitada, onde cada linha deverá ser multiplicada pelo maior elemento da linha em questão; mostre a matriz resultante.
Utilize apenas os fluxos de entrada e saída da linguagem C++ (cin e cout, respectivamente). **NÃO** utilize nem scanf, nem printf.

Adaptado de: Ascencio e Campos (2012, p. 250)

Exemplos:

Entrada	Saída
2 1 1 1	4 2 2 2
1 3 1 1	3 9 3 3
1 1 4 1	4 4 16 4
1 1 1 5	5 5 5 25
1 1 6 1	6 6 36 6
1 7 1 1	7 49 7 7
0 9 16 19	0 171 304 361
12 1 16 2	192 16 256 32
6 1 9 7	54 9 81 63
1 16 13 11	16 256 208 176
6 16 8 12	96 256 128 192
1 12 12 12	12 144 144 144
1 4 5 2	5 20 25 10
3 1 0 5	15 5 0 25
7 5 7 9	63 45 63 81
3 8 9 6	27 72 81 54
2 8 6 2	16 64 48 16
1 0 5 4	5 0 25 20
18 14 5 1	324 252 90 18
29 28 11 22	841 812 319 638
0 16 8 11	0 256 128 176
25 9 16 8	625 225 400 200
14 3 13 20	280 60 260 400
7 8 8 0	56 64 64 0

10. Usando a linguagem C++, escreva um programa que declare e use um registro (`struct`) para armazenar uma data (com dia, mês e ano):

- ler 5 datas, cada uma formada por 3 números inteiros correspondendo, respectivamente, a dia, mês e ano, armazenando-as em um registro (usando a estrutura declarada para armazenamento para datas) e mostrando para cada data lida a mensagem "+" (caso seja uma data válida) ou "-" (caso seja inválida);
- mostrar a data mais antiga entre as datas válidas, no formato "DD/MM/AAAA" (caso nenhuma data válida tenha sido fornecida, mostrar "**").

Utilize apenas os fluxos de entrada e saída da linguagem C++ (`cin` e `cout`, respectivamente). **NÃO** utilize nem `scanf`, nem `printf`.

Exemplos:

Entrada			Saída
15	4	2020	+
1	3	2021	+
31	9	2020	-
12	4	2020	+
29	2	2021	-
			12/04/2020
29	2	1600	+
29	2	2000	+
29	2	1996	+
29	2	2000	+
29	2	1800	-
			29/02/1600
28	2	1998	+
27	2	1998	+
29	2	1998	-
30	2	1998	-
29	2	2000	+
			27/02/1998
5	8	2020	+
3	9	2021	+
2	13	2020	-
1	10	2020	+
6	12	2021	+
			05/08/2020
15	0	2020	-
0	3	2021	-
31	4	2020	-
31	6	2020	-
31	11	2021	-
			*