

Lista de Exercícios 1 (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.c`, `Exercicio002.c`, ..., `Exercicio100.c`);
- 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`).

6. Escreva uma função em C, chamada `inverte_int`, que recebe um número inteiro positivo (por exemplo, 73483) e devolve outro inteiro que é este número revertido (ou seja, 38437). Você não sabe quantos dígitos o número terá e não pode receber um dígito de cada vez.

Acrescente a sua função ao código a seguir:

```
#include <stdio.h>

int main() {
    int n;
    scanf("%d", &n);
    printf("%d\n", inverte_int(n));
    return 0;
}
```

O código acima lê da entrada padrão um valor inteiro e mostra o resultado retornado pela função `inverte_int` para este valor.

Adaptado de uma questão da disciplina de *Introducao a Programacao - ECo* (do curso de Bacharelado em Engenharia de Computação, da PUCRS), elaborada pelo professor João Batista Olivera (agosto de 2022).

Exemplos:

Entrada	Saída
12345	54321
54321	12345
43210	1234
1020304	4030201
10203040	4030201
111222	222111
12121212	21212121
11110000	1111
79385	58397
12344321	12344321

7. Escreva um programa em C que lê 50 palavras da entrada padrão, de até 20 caracteres, determinando quantas palavras tem exatamente 7 caracteres.

Adaptado de uma questão da disciplina de *Introducao a Programacao - ECo* (do curso de Bacharelado em Engenharia de Computação, da PUCRS), elaborada pelo professor João Batista Olivera (agosto de 2022).

Exemplos:

Entrada	Saída
Alligator Anteater Armadillo Bat Bear Butterfly Camel Cat Chameleon Chicken Cow Dog Donkey Dolphin Elephant Fish Fox Frog Giraffe Goat Hen Hippopotamus Horse Kangaroo Lion Monkey Moose Mouse Ostrich Owl Ox Panther Peacock Pelican Penguin Pig Rhino Rooster Rabbit Shark Sheep Sloth Spider Squirrel Tiger Toucan Turtle Vulture Whale Zebra	10
Abelia Amaranthus Angelica Anthurium Bauhinia Begonia Buttonbush Calendula Camellia Celosia Cinquefoil Cornflower Cosmos Daffodil Dahlia Daisy Forsythia Foxglove Gardenia Geranium Gerbera Gladiola Gladiolus Graceland Gypsophila Hibiscus Hollyhocks Iris Jasmine Larkspur Lavender Lilac Lily Lotus Mallow Marigold Orchid Pansy Petunia Primrose Rhododendron Rose Snapdragon Speedwell Spirea Summersweet Sunflower Vinca Violet Zinnia	5
Ackee Apple Apricot Atemoya Avocados Banana Blackcurrant Blueberry Cantaloupe Cherry Coconut Cranberry Dates Dragonrfruit Durian Feijoa Fig Gooseberries Grapefruit Guava Hazelnut Honeyberries Jackfruit Jenipapo Kiwi Langsat Lime Lychee Mango Mangosteen Melon Mulberry Muskmelon Nance Nectarine Olive Oranges Papaya Peach Pear Physalis Pineapple Pomegranate Raspberries Soursop Starfruit Strawberries Sweetsop Tangerine Watermelon	6
Accordion Banjo Bassoon Bell Bugle Celesta Cello Clarinet Comet Cornet Cymbal Damru Dholak Drums Ektara Euphonium Flute Gong Gramophone Guitar Gu-zheng Harmonica Harmonium Harp Keyboard Lute Maracas Marimba Mridangam Oboe Oud Piano Piccolo Pungi Sarangi Sarod Saxophone Shehnai Sitar Spinet Tabla Tambourine Triangle Trombone Trumpet Tuba Ukulele Veena Violin Xylophone	9
us most day give these any want new even way well first work our how two use after back also think over its come only look now then than other see them could some good your year into people take know him just no time like can make when me	0
prolixo sublime empatia recesso refutar inferir cordial exortar emergir sucinto excesso conciso incauto parcial austero imputar trivial aspecto auferir estirpe excerto candura salutar padecer sensato alegria preciso fomento certeza mitigar diferir hesitar escasso vigente incitar intuito oriundo deleite definir exilado modesto sentido indagar almejar abrange deferir ademais arcaico mancebo virtude	50

8. Para esta questão, suponha que você tem as seguintes structs definidas em C:

```
struct motor {
    int cc, potencia, cilindros;
};

struct carro {
    char modelo[45];
    int ano;
    char transmissao; // M = manual, A = automatica
    int km;
    char combustivel; // F = flex, G = gasolina, D = diesel
    int numportas;
    struct motor m;
    double preco;
};
```

E também o seguinte vetor de carros:

```
struct carro revenda[20] = {
    {"Chevrolet_Celta_1.0_MPF_8V", 2015, 'M', 72660, 'F', 3, { 999, 70, 4}, 39900.0},
    {"Chevrolet_Onix_1.0_LT_8V_Hatch", 2018, 'M', 94094, 'F', 4, { 999, 82, 4}, 56900.0},
    {"Chevrolet_S10_2.8_LT_4x4_CD_16V_Turbo", 2018, 'A', 93649, 'D', 4, {2776, 200, 4}, 170900.0},
    {"Fiat_Argo_1.0_Drive_6V", 2020, 'M', 35994, 'F', 4, { 999, 77, 3}, 64900.0},
    {"Fiat_Palio_1.8_MPI_Adventure_Weekend_16V", 2014, 'M', 123738, 'F', 4, {1747, 132, 4}, 49900.0},
    {"Fiat_Siena_1.4_MPI_EL_8V", 2012, 'M', 137570, 'F', 4, {1368, 86, 4}, 34900.0},
    {"Fiat_Siena_1.6_MPI_Essence_16V", 2012, 'M', 152349, 'F', 4, {1598, 117, 4}, 36900.0},
    {"Fiat_Siena_1.0_MPI_EL_8V", 2011, 'M', 339237, 'F', 4, { 999, 75, 4}, 26900.0},
    {"Fiat_Uno_1.0_Attractive_8V", 2020, 'M', 80565, 'F', 4, { 999, 75, 4}, 50900.0},
    {"Fiat_Uno_1.4_Way_8V", 2014, 'M', 143131, 'F', 4, {1368, 85, 4}, 40900.0},
    {"Ford_Ecosport_1.5_SE_12V", 2018, 'M', 63866, 'F', 4, {1497, 137, 3}, 76900.0},
    {"Ford_Ecosport_2.0_Freestyle_4WD_16V", 2017, 'M', 118177, 'F', 4, {1999, 147, 4}, 74900.0},
    {"Ford_Ecosport_1.6_XLS_8V", 2011, 'M', 153671, 'F', 4, {1598, 111, 4}, 43900.0},
    {"Honda_Civic_1.8_LXS_Sedan_16V", 2007, 'A', 124170, 'G', 4, {1799, 140, 4}, 46900.0},
    {"Honda_HR-V_1.8_EXL_Flexone_16V", 2016, 'A', 58820, 'F', 4, {1799, 139, 4}, 95900.0},
    {"Hyundai_HB20_1.6_Comfort_Style_16V", 2018, 'M', 71299, 'F', 4, {1591, 128, 4}, 65900.0},
    {"Hyundai_IX35_2.0_MPI_4x2_16V", 2012, 'A', 140865, 'G', 4, {1998, 178, 4}, 70900.0},
    {"Hyundai_SantaFe_3.3_GLS_4x4_V6_TripTronic", 2018, 'A', 95394, 'G', 4, {3342, 270, 6}, 156900.0},
    {"Hyundai_Sonata_2.4_GLS_16V", 2012, 'A', 134467, 'G', 4, {2359, 182, 4}, 69900.0},
    {"Jeep_Renegade_1.8_Sport_4x2", 2016, 'M', 93352, 'F', 4, {1747, 130, 4}, 76900.0}
};
```

E escreva um programa em C que leia, respectivamente, o tipo de combustível, o número de cilindros e o valor máximo de um carro e conte quantos carros da revenda funcionam com esse combustível, tem esse número de cilindros e tem preço menor ou igual ao valor máximo fornecido.

Adaptado de uma questão da disciplina de *Introdução a Programação - ECO* (do curso de Bacharelado em Engenharia de Computação, da PUCRS), elaborada pelo professor João Batista Olivera (agosto de 2022).

Exemplos:

Entrada	Saída
F 4 30000.0	1
F 4 34900.0	2
F 4 36900.0	3
F 4 40000.0	4
G 6 100000.0	0
G 6 156900.0	1
G 6 157000.0	1
F 4 100000.0	13
D 4 200000.0	1
F 3 80000.0	2

9. Escreva um programa em C que recebe um inteiro positivo e diz quantos “33” existem no inteiro. Por exemplo 331, 33533 e 333 tem 1, 2 e 2 ocorrências de “33”, respectivamente.

Adaptado de uma questão da disciplina de *Introdução a Programação - ECO* (do curso de Bacharelado em Engenharia de Computação, da PUCRS), elaborada pelo professor João Batista Olivera (agosto de 2022).

Exemplos:

Entrada	Saída
331	1
33533	2
333	2

10. Escreva um programa em C que leia da entrada padrão um vetor A com 50 valores inteiros e determine o tamanho da maior sequência de valores ímpares que estão lado a lado dentro desse vetors.

Adaptado de uma questão da disciplina de *Introducao a Programacao - ECo* (do curso de Bacharelado em Engenharia de Computação, da PUCRS), elaborada pelo professor João Batista Olivera (agosto de 2022).

Exemplos:

Entrada	Saida
2 2 3 5 7 2 2 2 3 9 11 13 15 21 33 2 2 2 2 2 4 1 3 5 7 4 4 4 4 4 11 13 15 11 13 15 11 13 15 1 3 5 7 9 1 3 5 7 9 1	20
8 4 2 2 10 8 2 2 2 10 4 2 8 10 10 4 8 6 2 2 4 8 8 4 10 6 10 6 8 10 8 8 2 6 6 4 4 4 2 8 4 4 10 6 10 6 2 8 8 10	0
8 4 2 1 10 8 2 2 2 10 4 3 8 10 10 5 8 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 4 4 4 1 8 4 4 10 7 10 6 2 8 9 10	1
8 4 2 1 10 8 2 2 2 10 1 1 8 10 10 1 8 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 4 4 1 1 8 4 4 10 7 10 6 2 8 9 1	2
8 4 2 1 1 1 2 2 2 10 1 1 1 10 10 1 1 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 4 4 1 1 8 4 4 10 7 10 6 2 1 9 1	3
8 4 1 1 1 1 2 2 2 10 1 1 1 10 10 1 1 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 4 1 1 1 8 4 4 10 7 10 6 1 1 9 1	4
8 1 1 1 1 1 2 2 2 10 1 1 1 1 10 1 1 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 1 1 1 1 8 4 4 10 6 1 1 1 1 9 2	5
1 1 1 1 1 1 2 2 2 10 1 1 1 1 1 2 1 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 1 1 1 1 8 4 4 10 1 1 1 1 1 9 2	6
1 1 1 1 1 1 3 2 2 10 1 1 1 1 1 5 1 6 2 2 4 8 8 7 10 6 10 6 8 10 8 9 2 6 6 1 1 1 1 8 4 4 5 1 1 1 1 1 9 2	7
11 3 9 5 5 3 3 7 3 3 5 7 7 7 3 11 9 9 7 7 3 7 9 7 7 3 11 5 9 5 9 9 7 7 5 9 5 9 9 7 11 7 3 3 3 5 11 5 9 3	50