

Instruções

Os problemas marcados com ▷ são os exercícios selecionados e que devem ser entregues no prazo determinado.

Nas listas de exercícios, em alguns há exemplos. Para deixar mais claro o que é entrada do usuário e o que é saída do programa, costumo colocar a entrada em **negrito** e a saída em fonte normal.

Prazo dos exercícios selecionados: 14/11/2021

Lista 3 (semana 4)

1. Escreva um programa em C que leia um inteiro **n** e gere os cinquenta primeiros termos da série:
 $1 + n, 2 * n, 3 + n, 4 * n, \dots$

- Exemplo (reticências utilizadas para abreviar o exemplo):

2
3 4 5 8 7 12 ... 100

2. Escreva um programa em C para ler um número inteiro k e imprimir os k primeiros números ímpares positivos e sua soma (não utilize fórmula fechada para calcular a soma).

- Exemplo:

4
1 3 5 7 16

3. Um número inteiro positivo, n , é dito **perfeito** se a soma de seus divisores, excetuando-se o próprio número, é igual ao próprio número. Por exemplo, 6 é um número perfeito, já que $6 = 1 + 2 + 3$. Escreva um programa em C que lê um número e escreva “Perfeito” caso seja um número perfeito e “Não perfeito”, caso contrário.

- Exemplo 1:

6
Perfeito

- Exemplo 2:

5
Não perfeito

4. Escreva um programa em C que imprima a tabela abaixo:

Número	Quadrado	Cubo	Raiz Quadrada	Raiz Cúbica
1.5	2.25	3.375	1.2247	1.1447
2.5	6.25	15.625	1.5811	1.3572
...
99.5	9900.25	985074.875	9.9749	4.6338

5. ▷ Escreva um programa em C para ler a idade de um grupo de 20 pessoas. Ao final, o programa deve determinar a pessoa mais idosa, a mais jovem, e a média das idades do grupo de pessoas com duas casas decimais de precisão.

- Exemplo 1:

62 43 41 36 53 80 25 65 74 77 22 39 78 59 48 60 23 31 9 61
80 9 49.30

- Exemplo 2:

61 63 29 32 32 70 91 28 50 63 14 61 92 51 67 23 51 33 78 47
92 14 51.80

6. Um número inteiro positivo, n , é dito **triangular** se, e somente se, ele é o resultado do produto de três

números inteiros positivos e consecutivos. Por exemplo, 24 é triangular, pois $24 = 2 \times 3 \times 4$. Escreva um programa em C que leia um número inteiro positivo, n , e escreva como saída “é triangular” se n for triangular e “não é triangular” caso contrário.

7. Adedonha é um popular jogo, principalmente entre as crianças, em que os jogadores precisam escrever nomes dentre diversas categorias (nome de pessoa, cidade, fruta, país, etc) que começam com determinada letra. Por exemplo, se a letra for P, o jogador pode escrever Pedro (nome de pessoa), Porto Alegre (cidade), Pêra (fruta), Paraguai (país). Para determinar a letra de cada rodada, uma possibilidade é cada jogador, após um “já!”, mostrar as mãos com uma certa quantidade de dedos levantados. Soma-se a quantidade de dedos levantados em todas as mãos e o resultado é convertido para letra, de tal forma que 1 corresponde à letra A, 2 à letra B, 3 à letra C, 4 à letra D e assim por diante. Após 26, que corresponde à letra Z, volta-se para a letra A.

Quando há muitos participantes, fazer a conversão pode ser uma tarefa trabalhosa. Para facilitar a brincadeira, escreva um programa em C que leia um número inteiro n representando a quantidade de participantes e, em seguida, escreva uma tabela com todas as conversões possíveis. Como cada pessoa pode levantar até 10 dedos, então o número máximo de conversões é **10 vezes n** . Por exemplo, para $n = 3$, o programa deve escrever (as reticências foram colocadas aqui somente para abreviar o exemplo):

- Exemplo:

```

3
1 A
2 B
...
26 Z
27 A
28 B
29 C
30 D

```

8. Considerando o jogo Adedonha anterior, escreva um programa em C que leia o mesmo número inteiro n , seguido de n números inteiros, cada um representando a quantidade de dedos levantados por cada um dos participantes. Depois o programa deve escrever na tela qual a letra da rodada. Caso o total seja 0, o programa deve escrever “Sortear novamente” e solicitar um novo conjunto de n números até que a soma seja diferente de zero. Por exemplo:

- Exemplo 1:

```

3 8 10 10
B

```

- Exemplo 2:

```

3 3 7 0
J

```

- Exemplo 3:

```

4 0 0 0 0
Sortear novamente
0 0 0 0
Sortear novamente
0 0 1 0
A

```

9. ▷ Você está com o seguinte problema: precisa dobrar uma folha na metade algumas vezes para que caiba em seu bolso. Crie um programa que leia o comprimento, a largura da folha e o comprimento do bolso (declare como float). Deve, em seguida, calcular quantas vezes será necessário dobrar a folha para que caiba no bolso (um dos lados da folha dobrada deve ser **menor** que o comprimento do bolso). Para facilitar a dobra, considere que você **sempre dobra a folha para reduzir o maior dos lados**.

Por exemplo: a folha possui tamanho 30.0 por 20.0 e a largura do bolso é 8. Primeiro a folha é dobrada

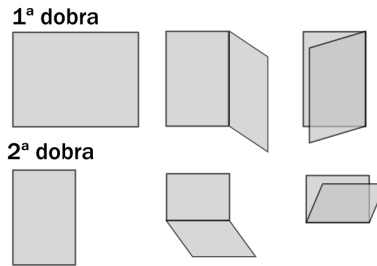


Figure 1: A primeira dobra foi no comprimento, a segunda dobra na largura

para 15 por 20 (pois 30 era o maior lado). Na segunda dobra, para 15 por 10 (pois 20 era o maior lado após a primeira dobra). Na terceira dobra, para 7.5 por 10 e, então, a folha cabe no bolso pois um dos lados é menor que 8. Portanto, o programa deve escrever 3 dobras.

10. “Amigo é coisa pra se guardar” – como diz Milton Nascimento. Os números também podem ser amigos entre si!

Para saber se dois números são amigos verifica-se se a soma dos divisores próprios de um é igual ao outro e vice-versa. Por exemplo: 220 e 284 são amigos pois os divisores próprios de 220 são 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 e 110, cuja soma é 284; e os divisores de 284 são 1, 2, 4, 71 e 142, cuja soma é 220. Sendo $D(x)$ a soma dos divisores próprios, então dois números inteiros A e B são amigos se, e somente se, $D(A) = B$ e $D(B) = A$.

O problema é que os números amigos são raros! Os primeiros números amigos são (220, 284), (1184, 1210), (2620, 2924).

O seu programa deve ler dois inteiros (suponha que sejam sempre diferentes) e escrever S caso sejam amigos e N caso contrário.

11. Escreva um programa em C para calcular uma aproximação para π . A aproximação pode ser obtida de $\pi = 4 - 4/3 + 4/5 - 4/7 + 4/9 - 4/11 + \dots$. O programa deve encerrar o processamento quando a variação no valor calculado for inferior a 0.0001.
12. Escreva um programa para calcular uma aproximação para $\cos(x)$, onde x é um número real digitado pelo usuário. A aproximação pode ser obtida de $\cos(x) = 1 - x^2/2! + x^4/4! - x^6/6! + \dots$. O programa deve encerrar o processamento quando a variação no valor calculado for inferior a 0.001.
13. \triangleright Escreva um programa em C que leia um inteiro x e escreva na tela o menor inteiro positivo n tal que $n < 46000$ e $n^2 + 3n + 5$ seja divisível por x . Caso não seja possível encontrar tal número, o programa deve escrever ‘impossível’.