

Lista de Exercícios 1 (C++)

Instruções

- As questões a seguir devem ser resolvidas usando a linguagem C++.
- Para cada questão, escreva o código em um arquivo .cpp (se você resolver todas as questões, devem existir 15 arquivos .cpp).
- Coloque sua identificação - nome e TIA - no início do arquivo, como comentário (use // no começo de cada linha que queira comentar).
- Inclua como comentário quaisquer referências (livros, artigos, sites, entre outros) usadas para responder as questões.
- Para as questões 11 a 15, que pedem para escrever funções, inclua um código que exemplifique o uso correto das funções. Por exemplo, na main() do seu código você chama a função que criou para a questão.
- **Entrega:** via link do Moodle até 04/03/2021 23:59.

Questões

1. Escreva um algoritmo que leia dois números informados pelo usuário. Cada número deve ser salvo em variáveis distintas, por exemplo, valor1 e valor2. Após ler os dois números, o algoritmo deve trocar o conteúdo das variáveis.
Exemplo: valor1 armazena o valor 30 e valor2 armazena o valor 50. Após o algoritmo ser executado, a variável valor1 deve armazenar o valor 50 e a variável valor2 deve armazenar o valor 30.
2. Escreva um algoritmo que leia a distância (em Km) e o tempo (em horas) que um veículo percorreu entre um ponto A e um ponto B. Em seguida, indique a velocidade média do veículo durante o trecho percorrido e se a velocidade média ficou acima do limite de 90 Km/h.
3. Escreva um algoritmo que leia uma letra e indique se é uma vogal ou consoante.
4. Escreva um algoritmo que leia um número e indique se o número é par ou ímpar.
5. Assumindo a equação de reta $3x + y = 8$, escreva um algoritmo que leia os valores para x e y e indique se o ponto (x, y) pertence à reta.
6. Escreva um algoritmo que calcule e exiba o fatorial de um número informado pelo usuário.
7. Escreva um algoritmo que leia 10 números inteiros e, após a leitura dos números, informe qual é o maior e qual é o menor.
8. Escreva um algoritmo que exiba a sequência de Fibonacci com N números, sendo N um número informado pelo usuário. A sequência de Fibonacci começa com 1 e

1 e os números subsequentes são obtidos com a soma dos dois números anteriores:
1, 1, 2, 3, 5, 8, 13, ...

9. Escreva um algoritmo que leia 10 números inteiros e, em seguida, exiba-os na ordem inversa que foram inseridas pelo usuário.
10. Escreva um algoritmo que usa um vetor para armazenar 6 números inteiros informados pelo usuário. Em seguida, ordene os elementos do vetor em ordem crescente. Por exemplo, caso o vetor contenha os números { 7, 3, 10, 5, 2, 6 } (nessa ordem), ao final do código o vetor deve armazenar { 2, 3, 5, 6, 7, 10 } (números em ordem crescente).
11. Escreva uma função `MulViaAdd()` que recebe dois números como parâmetros. O primeiro parâmetro pode ser um número real e o segundo parâmetro deve ser um número inteiro. A função deve retornar a multiplicação entre os parâmetros, porém, a multiplicação deve ser feita usando apenas a operação de adição.
12. Escreva uma função `Min()` que recebe dois parâmetros numéricos e retorna o menor valor entre eles.
13. Escreva uma função `Max()` que recebe dois parâmetros numéricos e retorna o maior valor entre eles.
14. Escreva uma função `VowelsCount()` que recebe uma string como parâmetro e retorna a quantidade de vogais encontradas na string. Considere que a string é um vetor de caracteres e use o `[]` para acessar cada caractere (elemento do vetor).
15. Dois círculos se sobrepõem (colidem) se a soma dos seus raios é maior que ou igual a distância entre seus centros. A distância entre dois pontos pode ser calculada usando o teorema de Pitágoras ($c = \sqrt{a^2 + b^2}$). Escreva uma função `AreCirclesColliding()` que recebe seis parâmetros: a posição `p1(x,y)` e o raio `r1` do primeiro círculo e a posição `p2(x,y)` e o raio `r2` do segundo círculo. Com os parâmetros informados, a função deve retornar `true` caso os círculos estejam sobrepostos ou `false` caso contrário.