

Classes e Objetos

1. Crie uma classe **Data** com três atributos inteiros: **dia**, **mês** e **ano**. Faça um construtor que inicializa as três variáveis e suponha que os valores passados serão corretos. A classe deve possuir um método para exibir a data em formato de números separados por barra: dia/mes/ano e outro método para exibir a data por extenso (ex: 12 de janeiro de 2015).
2. Crie uma classe **Rectangle** com atributos **length** e **width**, cada um dos quais assume o padrão de 1. Forneça funções-membro que calculam os atributos **perimeter** e **area** do retângulo. Além disso, forneça as funções **set** e **get** para os atributos **length** e **width**. As funções **set** devem verificar se **length** e **width** são números de ponto flutuante maiores que 0,0 e menores que 20,0.
3. Implemente em C++ uma classe chamada **Aquecedor**. Ela deve ter um único atributo chamado **temperatura**, cujo tipo deve ser um ponto flutuante de precisão dupla. Defina um construtor que não recebe parâmetros e inicializa a temperatura em 20 graus. Crie os métodos **aquecer** e **resfriar** que aumentam e diminuem a temperatura em 5 graus, respectivamente. Defina um método para retornar o valor da temperatura.
4. Altere a classe do exercício anterior para que ela tenha três novos atributos: temperatura mínima, temperatura máxima e fator de incremento da temperatura. Os dois primeiros devem ser inicializados com 10 e 40 graus respectivamente no construtor. A classe deve ter um construtor sem parâmetros, que definirá o fator de incremento em 5 graus, um segundo construtor que recebe a temperatura inicial e um terceiro que recebe a temperatura inicial e o fator de incremento.

Altere os métodos existentes na classe de forma apropriada com o objetivo de manter o estado do objeto sempre válido (ex: o fator de incremento deve ser usado toda vez que os métodos **aquecer** e **resfriar** forem chamados). Escreva mensagens na saída padrão quando uma ação não puder ser executada por não ser um estado de objeto válido.

Por fim, crie um método que permita alterar o fator de incremento da temperatura depois de um objeto já ter sido criado.
5. Crie uma classe **SavingsAccount**. Utilize um membro de dados **static annual_interest_rate** para armazenar a taxa de juros anual para cada um dos correntistas. Cada membro da classe contém um membro de dados **private savings_balance** para indicar a quantia que os correntistas têm atualmente em depósito. Forneça a função-membro **calculate_monthly_interest** que calcula os juros mensais multiplicando o **balance** [saldo] pelo **annual_interest_rate** dividido por 12; esses juros devem ser adicionados a **savings_balance**. Forneça uma função-membro **static modify_interest_rate** que configura o **static annual_interest_rate** com um novo valor.

Referências

- [1] D.J. Barnes and M. Kölling. *Programação orientada a objetos com java: uma introdução prática usando Blue J*. PRENTICE HALL BRASIL, 2004.
- [2] P.J. Deitel and H.M. Deitel. *C++ how to Program*. Deitel series. Pearson Prentice Hall, 2010.
- [3] G.L. McDowell. *Cracking the Coding Interview: 189 Programming Questions and Solutions*. CareerCup, LLC, 2015.