



Instituto Federal de Educação, Ciência e Tecnologia do Piauí
Campus Teresina Zona Sul
Licenciatura em Informática
Disciplina: Programação Orientada a Objetos

Exercício Pré-Estático

Questão 1

Crie um programa que controla as senhas da fila do caixa de um banco. Neste sistema, o caixa deve possuir um número, a última senha gerada e a senha que está em atendimento. Além disso, ele também tem um construtor, um método para gerar senha e outro para chamar senha.

O método de gerar senha, deve **retornar** um número simulando a senha que será entregue ao cliente, para que ele espere a sua vez na fila. Assim, ele deve incrementar o valor da última senha gerada e retornar esse valor. O método chamar senha, deve retornar a próxima senha que será atendida pelo caixa. Dessa forma, ele deve incrementar o valor da senha em atendimento e retornar esse valor. O diagrama da classe Caixa, pode ser visto abaixo. Nele observamos todos atributos e métodos da classe.

Caixa
- numero: int - ultimaSenhaGerada: int - senhaEmAtendimento: int
+ Caixa (numero:int) + getNumero(): int + gerarSenha(): int + chamarSenha(): int

Por último, simule a utilização deste caixa criando a classe **Executar**. Nela, crie o método *main* e nele crie um caixa, faça este caixa gerar senhas e por fim, chamar as senhas geradas.

Questão 2

No sistema da questão anterior existe um erro grave. O caixa pode chamar senhas sem nem ter gerado elas. Por exemplo, vamos supor que o caixa gere as senhas 1 e 2. Após isso, ele chama as senhas 1, 2 e 3. Observe que a senha 3 ainda não foi gerada, mas, no sistema atual, nada impede que isso ocorra. Sabendo disso, altere o programa anterior para solucionar este problema.

Questão 3

Outro defeito do sistema anterior é que as senhas geradas não são armazenadas em nenhum local. Existe apenas um controle de qual foi a última senha gerada, mas todas as demais não são conhecidas. Para melhorar esse sistema, seria necessário que, as senhas geradas fossem adicionadas a uma estrutura do tipo fila (FIFO). Em java, uma fila pode ser simulada através do objeto ArrayList. No nosso livro texto, o ArrayList é explicado na página 225 a 227, mas também pode ser visto neste [link](#). Sabendo disso, altere a classe Caixa, para que as senhas geradas sejam armazenadas em uma fila e, a medida que forem chamadas para atendimento, elas devem sair da fila. Por fim, crie um método listarSenhas, que retonar todas as senhas geradas e não

atendidas. A nova classe Caixa pode ser vista no diagrama abaixo. Observe que nele o método *listarSenhas* retorna uma *ArrayList*, que é justamente a fila de espera.

Caixa
- numero: int - fila: ArrayList - senhaEmAtendimento: int
+ Caixa (numero:int) + getNumero(): int + gerarSenha(): int + chamarSenha(): int + listarSenhas(): ArrayList

Desafio

Atualmente, é muito comum um caixa ter duas filas, uma normal (sem preferência) e outra preferencial. Altere o sistema anterior para que este caixa possa ter uma fila normal e outra preferencial, ao mesmo tempo. Um detalhe que pode ajudar é que, agora, a classe deve ter dois métodos de distribuir senhas, um para senhas da fila normal e outro para a preferencial. No momento de chamar a próxima senha, o retorno também deve informar se a senha chamada é preferencial ou normal. Para isso, possa ser que o método de chamar senha retorne uma ***String*** no lugar de um ***int***. Um exemplo do novo caixa pode ser visto no diagrama abaixo. Nele, os atributos foram suprimidos de propósito, para não direcionar o aluno a uma única solução.

Caixa
...
+ Caixa (numero:int) + getNumero(): int + gerarSenhaNormal(): int + gerarSenhaPreferencial(): int + chamarSenha(): String ...