

Ativar e desativar caixas ociosos

Foram acrescentados os seguintes parâmetros à nossa simulação:

- Número máximo de clientes na fila para ativar um caixa ocioso
- Tempo máximo que um caixa pode ficar livre para ficar ocioso

Com este controle, podemos ativar mais caixas se a demanda estiver alta, e desativar caixas quando a demanda estiver baixa.

Temos então dois novos métodos na classe `Agencia` que são:

- `AtivarCaixa()`
- `DesativarCaixa()`

Agora analise a lógica abaixo, do *loop* da simulação:

```
For I As Integer = 1 To TempoMaximoSimulacao
    AgenciaBancaria.AtivarCaixa(Acrescenta)
    If vContadorMinuto = 1 Then
        Dim vNumClientes As Integer = SerieHistoricaNumClientes.getNumeroAleatorio
        For J As Integer = 1 To vNumClientes
            Dim vTempoServico As Integer = SerieHistoricaServico.getNumeroAleatorio
            AgenciaBancaria.EntrarNaFila(vTempoServico)
            System.Threading.Thread.Sleep(100)
        Next
    End If
    vContadorMinuto += 1
    If vContadorMinuto = 60 Then
        vContadorMinuto = 1
    End If
    AgenciaBancaria.ExecutaAtendimento(1)
    AgenciaBancaria.DesativarCaixa(Retira)
    Me.Refresh()

    System.Threading.Thread.Sleep(TempoPasso)
Next
```

Em que momento devemos utilizar os métodos `AtivarCaixa()` e `DesativarCaixa()` ?

Selecione uma alternativa

A Logo após o `For` :

```
For I As Integer = 1 To TempoMaximoSimulacao
    AgenciaBancaria.AtivarCaixa()
    AgenciaBancaria.DesativarCaixa()
```

No final do `For` :

B

```
    AgenciaBancaria.AtivarCaixa()  
    AgenciaBancaria.DesativarCaixa()  
Next
```

C

AtivarCaixa() no início do For e DesativarCaixa() no final:

```
For I As Integer = 1 To TempoMaximoSimulacao  
    AgenciaBancaria.AtivarCaixa()  
    (.... Código ....)  
    AgenciaBancaria.DesativarCaixa()  
Next
```