

Este teste tem a duração de **2 horas** e é **sem consulta**.

1. Como certamente é do seu conhecimento, o SMS é um serviço de pequenas mensagens de texto (até 160 caracteres), disponível entre telemóveis. Implemente a classe *SMS*. Esta classe deve providenciar uma implementação para representar estas mensagens de texto, partilhadas entre um remetente (nº de telemóvel) e um destinatário (outro número de telemóvel).
 - (a) Crie um construtor que dados dois números, remetente e destinatário, e um texto, cria o correspondente objecto do tipo *SMS*. Caso o texto tenha mais de 160 caracteres deve considerar os primeiros 160.
 - (b) Apresente implementações para os métodos:
 - i. selectores para saber o destinatário e a mensagem;
 - ii. o método modificador que permita alterar a mensagem, com o conteúdo da *String* passada por parâmetro
 - iii. *toString()*, que retorne uma representação do tipo
DE: 916236532
PARA: 965297563
Tou a tua espera
 - iv. o método *size()* que retorna o número de caracteres da mensagem
 - v. *equals(Object x)*, que retorne true se dois *SMS* são iguais.
 - vi. *clone()* que retorna um objecto exatamente igual ao objecto original mas com outra referência.
2. Implemente a classe *Gestor_Mensagens*. Um gestor de mensagens é responsável pela gestão das mensagens enviadas e recebidas por um telemóvel. Usualmente, um gestor contempla o número de telemóvel ao qual está associado e duas listas de mensagens, uma para as mensagens recebidas e outra para as mensagens enviadas. Implemente para esta classe:
 - (a) Um construtor que recebe um inteiro correspondente ao número de telemóvel do gestor e inicializa as listas de mensagens: enviadas e recebidas
 - (b) o método *void receber(SMS msg)* que adiciona a mensagem *msg* à lista das mensagens recebidas
 - (c) o método *void criar_msg(int destino, String texto)* que cria a mensagem correspondente e a coloca na lista de mensagens enviadas
 - (d) o método *String ler(int n)* que retorna a *String* correspondente à n-ésima mensagem da lista de recebidas. Caso não exista tal mensagem deve ser retornada a *String* vazia
 - (e) o método *void apagar(SMS msg)* que apaga a mensagem passada por parâmetro, da lista de mensagens recebidas
 - (f) o método *void limpar_recebidas()* que apaga toda a lista de mensagens recebidas

3. Como sabe é possível, actualmente, adicionar uma imagem, um som, ou um vídeo às mensagens. Que alterações introduziria na definição da classe SMS para contemplar estas novas mensagens com anexo? Considere que o anexo é uma String, correspondente ao nome do ficheiro que quer anexar.
- (a) Também as mensagens quando são recebidas são marcadas como "não lidas" na lista das mensagens recebidas, já as mensagens enviadas não têm este tipo de etiqueta, dado que não tem qualquer significado o "não lida" para este tipo de mensagens. Que alterações introduziria na classe/hierarquia de classes implementada para contemplar esta característica das mensagens recebidas?