

Exercícios POO

1 Desenvolvimento da aplicação *Agenda de Contactos*

1.1 Descrição do problema

O objectivo deste exercício é o desenvolvimento de uma aplicação para gestão de uma agenda de contactos (desenvolvido em IP). A aplicação deve gerir os contactos de uma pessoa. Cada *contacto* é caracterizado pelo *nome*, um número de *telefone* e um *email*. Assuma que tipicamente não há mais de 100 contactos registados na agenda. A aplicação deve permitir:

1. Adicionar um novo contacto na agenda (comando **AC**). São fornecidos o nome, o telefone e o email do contacto. Em caso de sucesso é apresentada a mensagem "Contact added." A operação falha se: (1) já existir um contacto com esse nome ("Contact already exists.").
2. Listar por ordem de inserção os contactos existentes na agenda (comando **LC**). A operação sucede sempre. Caso a agenda não tenha contactos é apresentada a mensagem "Contact book empty."
3. Consultar o telefone de um contacto (comando **GP**). É fornecido o nome do contacto e é devolvido o telefone desse contacto. A operação falha se: (1) o nome não corresponder a um contacto existente ("Contact does not exist.").
4. Consultar o email de um contacto (comando **GE**). É fornecido o nome do contacto e é devolvido o email desse contacto. A operação falha se: (1) o nome não corresponder a um contacto existente ("Contact does not exist.").
5. Actualizar o telefone de um contacto (comando **SP**). São fornecidos o nome do contacto e o novo número de telefone. Em caso de sucesso é apresentada a mensagem "Contact updated." A operação falha se: (1) o nome não corresponder a um contacto existente ("Contact does not exist.").
6. Actualizar o email de um contacto (comando **SE**). São fornecidos o nome do contacto e o novo email. Em caso de sucesso é apresentada a mensagem "Contact updated." A operação falha se: (1) o nome não corresponder a um contacto existente ("Contact does not exist.").
7. Remover um contacto da agenda (comando **RC**). É fornecido o nome do contacto. Em caso de sucesso é apresentada a mensagem "Contact removed." A operação falha se: (1) já não existir um contacto com esse nome ("Contact does not exist.").
8. Procurar um contacto dado o número de telefone (comando **GN**). É fornecido o número de telefone e é devolvido o nome do contacto com o número dado. Caso existam vários contactos com o mesmo número de telefone, é devolvido o contacto mais antigo. A operação falha se: (1) o número de telefone não corresponder a nenhum contacto ("Phone number does not exist.").

9. Verificar se existem contactos com o mesmo número de telefone (comando [EP](#)). A operação sucede sempre. Caso existam contactos com o mesmo número de telefone é apresentada a mensagem "There are contacts that share phone numbers.", caso contrário é apresentada a mensagem "All contacts have different phone numbers."
10. Terminar a execução do programa "Goodbye!"

Note que o código da aplicação disponibilizado no moodle já implementa a maioria das funcionalidades, por isso apenas terá que completar as funcionalidades em falta: procurar um contacto por telefone (comando [GN](#)) e verificar se existem telefones repetidos (comando [EP](#)).

1.2 Exemplo de interacção com a aplicação

Desenvolva a sua aplicação para que esta garanta o modelo de interacção ilustrado no exemplo seguinte (o caracter ↵ representa uma mudança de linha):

```
LC↵
Contact book empty.↵
↵
AC↵
Joana Dias↵
99999999↵
Joana@fct.unl.pt↵
Contact added.↵
↵
AC↵
Joana Dias↵
99999999↵
Joana@fct.unl.pt↵
Contact already exists.↵
↵
AC↵
Joana Horas↵
91999999↵
Joana@gmail.com↵
Contact added.↵
↵
RC↵
Joana Dias↵
Contact removed.↵
↵
RC↵
Joana Dias↵
Contact does not exist.↵
↵
AC↵
Joana Dias↵
99999999↵
Joana@fct.unl.pt↵
Contact added.↵
↵
GP↵
Joana Dias↵
99999999↵
↵
```

GP↵
Joana Meses↵
Contact does not exist.↵
↵
GE↵
Joana Dias↵
Joana@fct.unl.pt↵
↵
GE↵
Joana Meses↵
Contact does not exist.↵
↵
SP↵
Joana Dias↵
253253253↵
Contact updated.↵
↵
SP↵
Joana Meses↵
253253253↵
Contact does not exist.↵
↵
GP↵
Joana Dias↵
253253253↵
↵
SE↵
Joana Dias↵
JoanaEu@tu.ele.pt↵
Contact updated.↵
↵
SE↵
Joana Meses↵
JoanaEu@tu.ele.pt↵
Contact does not exist.↵
↵
GE↵
Joana Dias↵
JoanaEu@tu.ele.pt↵
↵
LC↵
Joana Horas; Joana@gmail.com; 91999999↵
Joana Dias; JoanaEu@tu.ele.pt; 253253253↵
↵
GN↵
253253253↵
Joana Dias↵
↵
GN↵
123456789↵
Phone number does not exist.↵
↵
EP↵
All contacts have different phone numbers.↵
↵

```
AC↵
Joana Minutos↵
253253253↵
Joana.Minutos@gmail.com↵
Contact added.↵
↵
EP↵
There are contacts that share phone numbers.↵
↵
Q↵
Goodbye!↵
↵
```

1.3 Desenvolvimento

Desenvolva a sua aplicação de acordo com as seguintes fases:

1. Submeta o código fonte da aplicação, tal como foi disponibilizado, ao *Mooshak* (leia as instruções no site do Moodle da disciplina).
2. De seguida implemente as funcionalidades em falta e teste a aplicação.
3. Submeta o código fonte da aplicação, após ter implementado as funcionalidades em falta, ao *Mooshak*.

Submissão ao Mooshak

Ao submeter o código de uma aplicação ao *Mooshak* irá obter uma das seguintes mensagens:

- **0 Compile Time Error**
 - o programa submetido causou um erro de compilação (e.g. código fonte mal compactado, código com erros de compilação) e foram obtidos 0 pontos.
- **0 Runtime Time Error**
 - o programa submetido causou um erro de execução (e.g. ciclo infinito, acesso a uma posição inválida de um vector) e foram obtidos 0 pontos.
- **0 Wrong Answer**
 - o programa falhou todos os testes e foram obtidos 0 pontos.
- **N Wrong Answer**
 - o programa passou alguns dos testes e foram obtidos N pontos (soma da pontuação dos testes que o programa passou).
- **N Presentation Error**
 - o programa passa todos os testes, mas difere em alguns testes no número de espaços e/ou mudanças de linhas. Foram obtidos N pontos.
- **100 Accepted**
 - o programa passou todos os testes e foram obtidos 100 pontos (pontuação máxima).

Ficheiros de testes

Os testes do Mooshak verificam de forma incremental a implementação dos vários comandos:

- Ficheiro de teste: `1_in_base.txt` (20 pontos)
Comandos testados: AC, LC, GP, GE, SP, SE, RC, Q
Contexto: são testadas as funcionalidade já implementadas no código disponibilizado no Moodle.
- Ficheiro de teste: `2_in_base_GN.txt` (40 pontos)
Comandos testados: todos os comandos do teste 1 e o comando GN
Contexto: são testadas as condições onde o comando GN pode falhar.
- Ficheiro de teste: `3_in_base_EP.txt` (40 pontos)
Comandos testados: todos os comandos do teste 1 e o comando EP