

## Lista 11

A empresa XYZ possui diversos funcionários, sendo que os dados dos funcionários (identificador, nome, sexo) são mantidos na lista **LISTA\_F**. Além disso, a empresa possui uma lista para cada departamento (Produção, Entregas, Atendimento cliente), as listas **LISTA\_P**, **LISTA\_E** e **LISTA\_A**, respectivamente. As listas **LISTA\_P**, **LISTA\_E** e **LISTA\_A** armazenam apenas os identificadores dos funcionários alocados em cada departamento.

Para otimizar o trabalho dos funcionários, a empresa permite que um funcionário seja alocado para mais de um departamento.

Escreva um programa em C++ para gerenciar as listas. As operações que o programa deve ter são listadas a seguir:

- `cadastrarFunc(IDF, Nome, sexo)` – adiciona um funcionário na lista **LISTA\_F**.
- `addFuncDep(IDF, IDD)` – com base no identificador do departamento informado pelo parâmetro **IDD** (0-Produção, 1-entregas, 2-Atendimento ao cliente), adiciona o funcionário **IDF** em uma das listas de departamento.
- `removeFunc(IDF)` – remove um funcionário da lista **LISTA\_F**.
- `mostraFunc()` – mostra o **nome** de todos funcionários da lista **LISTA\_F**.
- `mostraDep(IDD)` com base no identificador do departamento informado pelo parâmetro **IDD** (0-Produção, 1-entregas, 2-Atendimento ao cliente), mostra o **nome** de todos funcionários do departamento.
- `mostraFuncEmComum(IDD1, IDD2)` – mostra o nome dos funcionários em comum entre dois departamentos.

As operações citadas acima possuem as seguintes restrições:

- `cadastrarFunc` – Os identificadores **IDF** são únicos, ou seja, não há **IDF** repetidos. Os funcionários devem ser adicionados na lista **LISTA\_F** em ordem crescente de **IDF**. A lista não pode sofrer uma ordenação dos seus elementos.
- `addFuncDep` – Um funcionário pode estar em mais de um departamento, mas não pode repetir em um mesmo departamento. Se o parâmetro **IDF** for inválido, a operação deve ser cancelada. Se o parâmetro **IDD** for inválido, a operação deve ser cancelada.
- `removeFunc` – Se um funcionário for removido, ele também deve ser removido dos departamentos em que ele estava alocado. Se o parâmetro **IDF** for inválido, a operação deve ser cancelada.
- `mostraDep` – Se o parâmetro **IDD** for inválido, a operação deve ser cancelada.
- `mostraFuncEmComum` – Cancelar a operação caso um dos identificadores seja inválido.

### Entrada

O programa possui como entrada um arquivo texto com o nome *entrada.txt*. Um exemplo de um arquivo de entrada é ilustrado no quadro abaixo.

O arquivo armazena uma sequência de operações, sendo uma operação por linha. Além disso, cada operação é seguida de seus parâmetros, onde cada parâmetro é composto por uma única string e separados por espaços em branco.

```
cadastrarFunc 5 Joao M
cadastrarFunc 1 Maria F
cadastrarFunc 3 Pedro M
cadastrarFunc 8 Joana F
cadastrarFunc 10 Marcos M
cadastrarFunc 1 José M
mostraFunc
addFuncDep 0 20
```

```
addFuncDep 50 0
addFuncDep 1 0
addFuncDep 1 0
addFuncDep 3 0
addFuncDep 8 0
addFuncDep 10 2
addFuncDep 1 2
addFuncDep 8 2
mostraDep 5
mostraDep 2
mostraFuncEmComum 50 20
mostraFuncEmComum 0 20
mostraFuncEmComum 0 2
removeFunc 8
mostraDep 0
mostraDep 2
mostraFuncEmComum 0 2
```

### **Saída**

O programa deve imprimir na saída o resultado de cada operação especificada na entrada, um resultado por linha. Um exemplo de saída para a entrada mostrada na seção anterior é dado a seguir.

```
O funcionário Joao (5) foi cadastrado com sucesso!
O funcionário Maria (1) foi cadastrado com sucesso!
O funcionário Pedro (3) foi cadastrado com sucesso!
O funcionário Joana (8) foi cadastrado com sucesso!
O funcionário Marcos (10) foi cadastrado com sucesso!
O funcionário José não foi cadastrado, pois o código 1 já existe.
Funcionários: Maria, Pedro, Joao, Joana, Marcos
```

Departamento 20 inexistente.

Usuário 50 inexistente.

O funcionário Maria foi adicionado no departamento de produção.

O funcionário Maria já está no departamento de produção.

O funcionário Pedro foi adicionado no departamento de produção.

O funcionário Joana foi adicionado no departamento de produção.

O funcionário Marcos foi adicionado no departamento de Atendimento ao Cliente.

O funcionário Maria foi adicionado no departamento de Atendimento ao Cliente.

O funcionário Joana foi adicionado no departamento de Atendimento ao Cliente.

Departamento 5 inexistente.

Departamento Atendimento ao cliente: Maria, Joana, Marcos

Departamentos 50 e 20 inexistentes

Departamento 20 inexistentes

Funcionários em comum nos departamentos de Produto e Atendimento: Maria, Joana

O funcionário Joana foi removido.

Departamento Produção: Maria, Pedro

Departamento Atendimento ao cliente: Maria, Marcos

Funcionários em comum nos departamentos de Produto e Atendimento: Maria