



(Exercício 1)

A empresa XYZ Ltda. está informatizando o seu departamento de Recursos Humanos e precisa da sua ajuda nesta tarefa. Considere o arquivo chamado “funcionarios.dat”, um arquivo de registros, contendo para cada funcionário seu número, seu nível salarial e seu departamento.

Como a administração da empresa é feita em nível departamental é importante que no arquivo, os funcionários de cada um dos departamentos estejam relacionados entre si e ordenados sequencialmente pelo seu número. Devido às frequentes mudanças interdepartamentais no quadro de funcionários, não é conveniente reestruturar o arquivo a cada uma destas mudanças. Desta maneira, o arquivo poderia ser organizado da seguinte forma:

linha	numFunc	nível	departamento	proximo
0	123	7	1	5
1	8765	12	1	-1
2	9210	4	2	-1
3	2628	4	3	6
4	5571	8	2	-1
5	652	1	1	9
6	7943	1	3	-1
7	671	5	3	12
8	1956	11	2	11
9	1398	6	1	10
10	3356	3	1	1
11	4050	2	2	4
12	2468	9	3	3

Em um segundo arquivo, “departamentos.dat”, um arquivo de registros, tem-se as seguintes informações:

codDepto	nomeDepto	inicio
1	vendas	0
2	contabilidade	8
3	estoque	7
4	entrega	2

Assim, o primeiro funcionário do departamento de vendas é o registro 0 do arquivo de funcionários e os demais funcionários do mesmo departamento são obtidos seguindo o campo proximo. Ou seja, os funcionários do departamento de vendas são os funcionários nos registros: 0, 5, 9, 10 e 1. Os funcionários do departamento de contabilidade são os funcionários nos registros: 8, 11 e 4.

Faça um programa em C que realize as seguintes operações:

- admissão de novo funcionário;
- demissão de funcionário;
- mudança de departamento por um funcionário;
- consulta a todos os funcionários de um departamento;
- consulta individual de um funcionário.

Para estas operações devem ser lidas as informações:

- código do tipo da operação: 0 para fim, 1 para admissão, 2 para demissão, 3 para mudança de departamento, 4 para consulta a todos os funcionários de um departamento e 5 para consulta individual de um funcionário;
- número do funcionário;
- nível salarial (somente no caso de admissão);
- número do departamento ao qual o funcionário passa a pertencer (no caso de admissão e mudança);
- número do departamento do qual o funcionário foi desligado (no caso de demissão e mudança);
- dados necessários para as operações de consulta.

O programa deve escrever as seguintes informações:

- os valores iniciais lidos dos arquivos;
- para cada operação: o tipo da operação realizada, os dados da operação e a forma final dos dados (de funcionários e departamentos).

No final do programa novos arquivos “funcionarios.dat” e “departamentos.dat” são gerados com os dados atualizados.

Detalhamento:

- a quantidade máxima de funcionários é 20;
- a quantidade máxima de departamentos é 5;
- se a quantidade máxima for ultrapassada o programa deve dar uma mensagem de erro;
- se for requisitada a remoção ou mudança de um funcionário não existente no departamento especificado o programa deve dar uma mensagem de erro;
- quando for requisitada a inserção de um novo funcionário é preciso verificar se um funcionário com o mesmo número já existe;
- se o código de operação for inválido o programa deve continuar lendo um novo código até que ele seja 0 (zero), 1 (um), 2 (dois) 3 (três), 4 (quatro) ou 5 (cinco).

(Exercício 2)

Um dos formatos reconhecidos pelos computadores atuais para imagens é o padrão PGM (Portable GrayMap). As imagens estão armazenadas em um arquivo ASCII que pode ser editado normalmente em qualquer editor de textos, mas **quando abertas em aplicativos apropriados (por exemplo photoshop ou gimp) a imagem pode ser visualizada, como por exemplo, uma fotografia.** Aplicativos de manipulação de imagens podem carregar as informações do arquivo em memória e fazer diversos tipos de conversão, como por exemplo clareamento de imagens, zoom, entre outros.

O formato PGM

Um arquivo em formato PGM tem o seguinte conteúdo:

- a primeira linha contém um identificador, que é “P2” para formatos PGM;
- a segunda linha contém a largura e a altura de uma matriz, isto é, o número de colunas e de linhas;
- a terceira linha contém um número que indica o maior valor da matriz da parte que segue;
- o restante do arquivo contém uma matriz de elementos (bytes) que representam um pixel da imagem em tons de cinza.

A figura 1 mostra um exemplo de um arquivo que é uma imagem em PGM: a primeira linha tem “P2”, a segunda linha contém a dimensão da matriz (10 x 11, observe que por definição o número de colunas vem antes); a terceira linha contém o maior elemento (40) da matriz que constitui o restante do arquivo.

```
P2
11 10
40
40 5 5 5 5 5 5 5 5 40 0
5 20 20 5 5 5 5 5 5 5 5
5 5 20 5 5 5 0 0 0 0 0
5 5 20 20 5 5 20 20 0 0 5
5 5 5 5 5 5 0 20 0 0 0
5 5 5 5 5 5 0 20 20 0 5
5 5 5 5 11 11 11 0 0 0 0
5 5 5 5 20 20 11 5 5 5 5
5 5 5 5 11 20 11 5 5 5 0
40 5 5 5 11 20 20 5 5 40 5
```

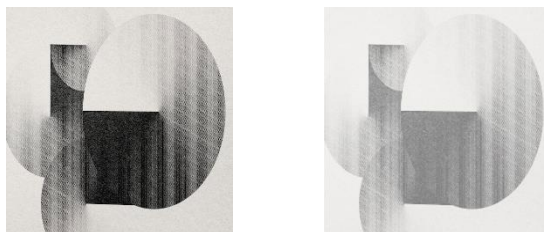
Figura 1: Exemplo de imagem no formato PGM.

Os valores da matriz são números entre 0 e 255, sendo que 0 representa o preto e 255 representa o branco. Os valores intermediários representam tons de cinza, quanto mais perto do 0 é um cinza cada vez mais escuro, quanto mais perto do 255 é um cinza bem claro. Na área de imagens estes números são chamados de pixels, que são na verdade um ponto da imagem quando exibidos em um aplicativo. Para visualizar as imagens é preciso abri-las em um programa visualizador de imagens PGM, como o Gimp por exemplo. Para mais detalhes: https://en.wikipedia.org/wiki/Netpbm#File_formats.

Sua tarefa consiste em ler um arquivo PGM (alguns arquivos estão disponíveis no endereço <https://people.sc.fsu.edu/~jburkardt/data/pgma/pgma.html>) e manipular a imagem correspondente, ou seja, realizar operações sobre a matriz que representa a imagem.

Operações a serem realizadas:

- Clarear a imagem: Um dos problemas mais simples que podemos resolver quando temos uma fotografia é o seu clareamento (somar um valor constante aos pixels) ou escurecimento (subtrair um valor constante aos pixels).



- Efeito de zoom: Ler uma imagem PGM e gerar uma outra com a metade do tamanho da original. Pode-se usar a técnica de reduzir a matriz original pela metade, observe o exemplo da figura 2. Cada bloco 2x2 dará origem a um único valor, que será a média dos 4 valores do bloco, resultando uma matriz 2x2. A motivação de se calcular a média é obter um tom de cinza que seja o mais próximo possível dos tons de cinza destes quatro elementos.

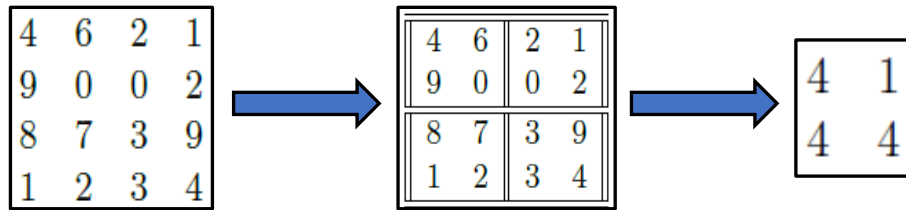
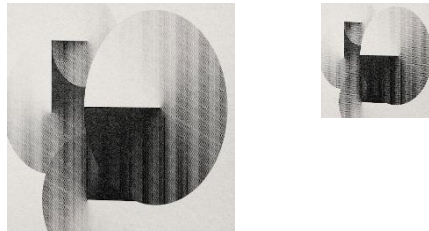
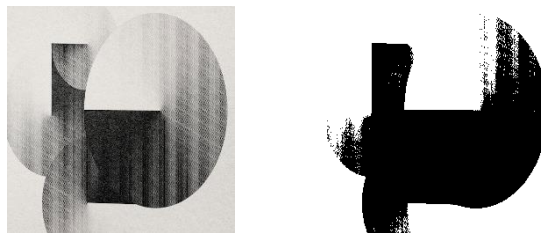


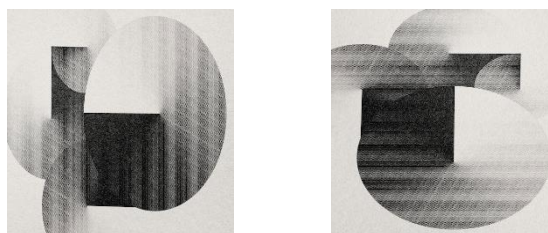
Figura 2. Redução do tamanho da matriz original pela metade.



- Binarizar a imagem: ler a imagem PGM em tons de cinza e transformá-la em preto e branco (onde 0 representa preto e 1 representa branco). Note que o limiar é obtido pelo valor médio dos pixels.



- Filtro de rotação simples: consiste em girar uma imagem em 90° no sentido horário.



Detalhamento:

- O nome do arquivo contendo a imagem PGM deverá ser fornecido na chamada do programa, ou seja, usar os parâmetros da função `main()`, `argc` e `argv`. Por exemplo, na chamada **C:\filtrosPGM.exe baloes.pgm**, o programa `filtrosPGM.exe` irá processar a imagem `baloes.pgm` e irá criar os arquivos resultantes `baloes1.pgm`, `baloes2.pgm`, `baloes3.pgm` e `baloes4.pgm`.
- Escrever uma função para cada manipulação na imagem.
- Gerar os arquivos de saída para cada manipulação.
- Usar uma struct para guardar as informações lidas do arquivo PGM (linha, coluna, maior e a matriz).