

Lista 6 - CI055 - Algoritmos e Estruturas de Dados I

A lista abaixo deve ser resolvida de maneira individual. As soluções dos exercícios deverão ser discutidas e entregues seguindo as instruções da página da disciplina:

- Alunos da turma do **Prof. Marcos Castilho** devem consultar:
<http://www.inf.ufpr.br/alexander/ci055/instrucoes-turma-marcos.html>
- Alunos da turma do **Prof. Daniel Weingaertner** devem consultar:
<http://www.inf.ufpr.br/alexander/ci055/instrucoes-turma-daniel.html>

O prazo para entrega desta lista é: **15 de Junho de 2015 (segunda-feira)**.

Enunciado do exercício

Esta lista possui um único exercício, o qual foi projetado para ser resolvido durante o período de uma semana. Ele adiciona características ao Exercício 3 da Lista 5 e, por isso, sua solução pode ser facilitada se for uma modificação da solução anteriormente elaborada:

http://www.inf.ufpr.br/alexander/ci055/lista_5.pdf

Tais características adicionais são muito semelhantes ao que consta no enunciado de problema no link abaixo, porém apenas um deslocamento horizontal e um vertical será permitido agora:

<http://br.spoj.com/problems/TV/>

Fazer um programa em Pascal para ler, da entrada padrão (*i.e.*, teclado), os dados de uma imagem de formato padrão PGM (Portable Gray Map). A leitura dos elementos da matriz deverá ser feita linha-a-linha (não por coluna).

O formato da impressão do programa, sempre na saída padrão (*i.e.*, monitor de vídeo), deverá ser exatamente o mesmo de uma imagem codificada no padrão PGM (Portable Gray Map), o qual terá $m+3$ linhas. A primeira linha só deverá possuir dois caracteres: P2. A segunda linha deverá ter dois inteiros: a quantidade de colunas da matriz seguido da quantidade de linhas. A terceira linha possui apenas o valor da constante max, a qual representa o valor da máxima intensidade de luz em uma análoga escala de cinza cujos valores variam de 0 a max. Veja um exemplo de arquivo PGM com $n = 4$ colunas, $m = 7$ linhas e $max = 12$.

```
cat texto_imagem_1.pgm
P2
4 3
12
6 7 8 5
10 11 12 9
2 3 4 1
```

Os deslocamentos horizontal e vertical da imagem devem ser lidos também da entrada padrão por meio de dois números inteiros. O primeiro valor representa o deslocamento na direção horizontal (valor positivo representa deslocamento para a direita e valor negativo para a esquerda). O segundo valor representa o deslocamento da direção vertical (valor positivo para cima e valor negativo para baixo). Esses dois valores constituem o final da leitura de dados do programa. Veja um exemplo de arquivo com dados de deslocamento horizontal de 3 pontos para a esquerda e vertical de 2 pontos para cima.

```
cat desloc_h_-3_v_2.txt
-3 2
```

Após a leitura, o programa deve efetuar ambos os deslocamentos sobre a matriz da imagem e, em seguida, imprimir a imagem completa na saída padrão no formato PGM. Por exemplo, se considerarmos os dois arquivos acima, o programa deverá imprimir os seguintes dados na tela.

```
cat texto_imagem_1.pgm desloc_h_-3_v_2.txt | ./sol_q1_lista6 | less
P2
4 3
12
1 2 3 4
5 6 7 8
9 10 11 12
```

Tal como já foi explicado anteriormente, também é possível visualizar a imagem PGM gerada pelo programa usando o utilitário display. Tente copiar a imagem PGM que está no link:

http://www.inf.ufpr.br/cursos/ci055/imagem_Algoritmos.pgm

Execute o utilitário display para mostrar a imagem graficamente, assim:

```
display imagem_Algoritmos.pgm &
```

Veja na Figura 1 como a imagem aparece na sua tela:



Figura 1: Imagem sem deslocamento apresentada com o utilitário display.

Crie também um exemplo de arquivo com dados de deslocamento horizontal de 48 pontos para a direita e vertical de 23 pontos para baixo. Veja como ele fica ao ser listado:

```
cat desloc_h_48_v_-23.txt
48 -23
```

Agora, provoque a entrada de dados por meio do utilitário cat aplicado, na ordem de entrada, aos dois arquivos. Canalise a saída do cat como entrada para o programa construído e, em seguida, canalise a saída para a execução do utilitário display por último para mostrar a imagem, assim:

```
cat imagem_Algoritmos.pgm desloc_h_48_v_-23.txt | ./sol_q1_lista6 | display &
```

Veja na Figura 2 como a imagem aparece na sua tela:

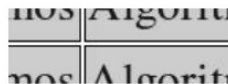


Figura 2: Imagem sem deslocamento apresentada com o utilitário display.

Os arquivos citados nos exemplos acima podem ser obtidos nos links abaixo:

```
http://www.inf.ufpr.br/cursos/ci055/imagem\_Algoritmos.pgm
http://www.inf.ufpr.br/cursos/ci055/desloc\_h\_48\_v\_-23.txt
http://www.inf.ufpr.br/cursos/ci055/texto\_imagem\_1.pgm
http://www.inf.ufpr.br/cursos/ci055/desloc\_h\_-3\_v\_2.txt
```

Uma possível composição do corpo de comandos do Programa Principal poderia ser o seguinte (note que a execução dos dois comandos write foi inibida por meio de comentário de código):

```
begin
  le_matriz(mat, m, n, max);
  (* write('Entre com o deslocamento Horizontal da imagem (negativo se for para a esquerda): '); *)
  read(deltah);
  (* write('Entre com o deslocamento Vertical da imagem (negativo se for para baixo): '); *)
  read(deltav);
  desloca_matriz(mat, m, n, deltah, deltav);
  imprime_matriz(mat, m, n, max);
end.
```

Finalmente, como sugestão, tente usar o software gimp para criar imagens e salvar em formato PGM para testar seu programa. Atenção ao salvar o arquivo para que a opção ASCII seja adotada (não utilize a opção raw).