

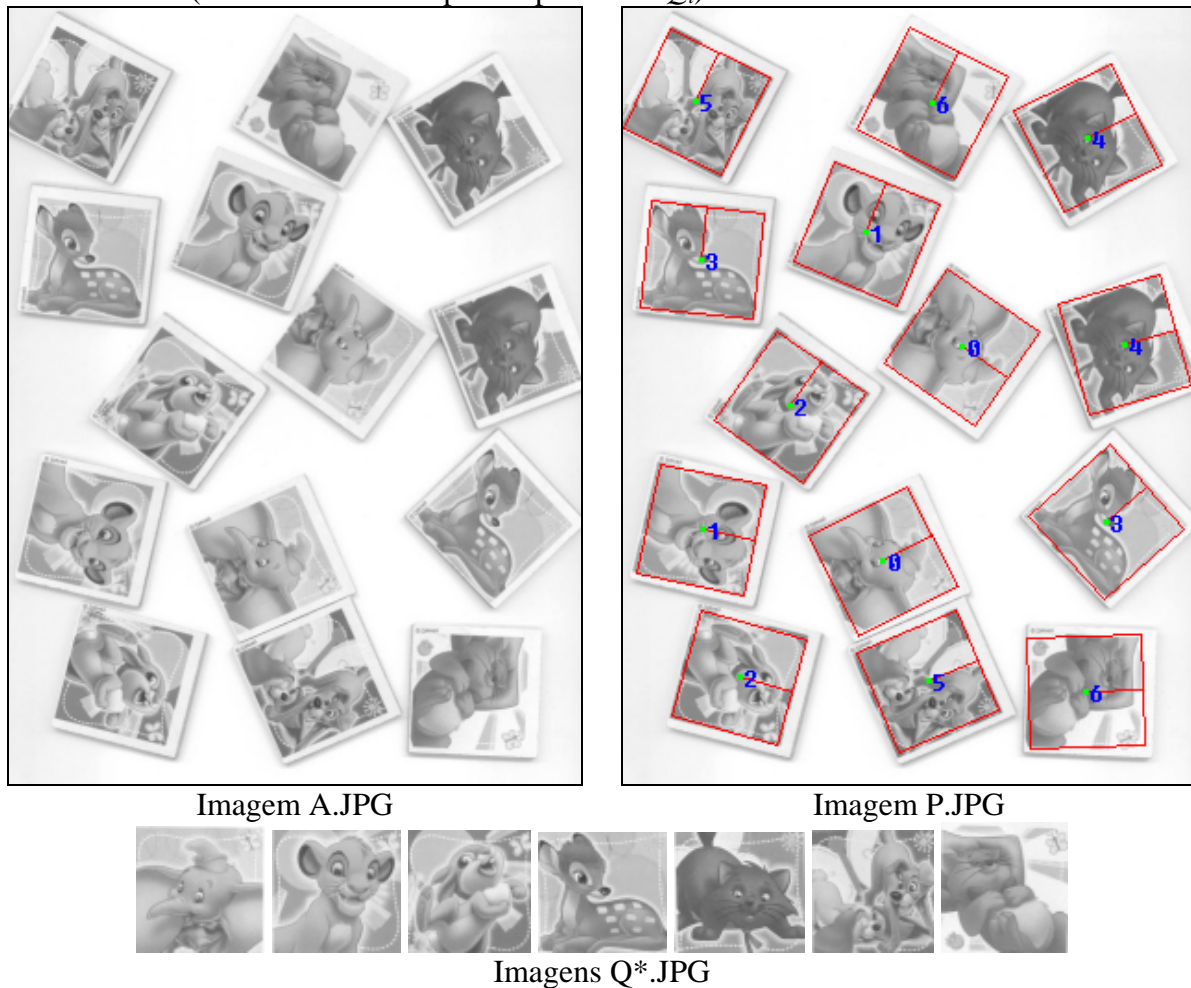
Faça o programa **EP1.CPP** em C/C++ que, lê uma imagem em níveis de cinzas  $A$  e procura nela uma imagem-máscara  $Q$ , gerando uma imagem  $P$  com o resultado da busca.

**C:>ep1 a.jpg p.jpg q.jpg**

Opcionalmente, pode fazer um programa que busca  $N$  imagens-máscaras  $Q_1, \dots, Q_N$  de uma só vez:

**C:>ep1 a.jpg p.jpg q\*.jpg**

Neste caso, o programa pode ficar mais rápido, pois parte do processamento pode ser feito uma única vez (em vez de ter de repeti-lo para cada  $Q_i$ ).



### Observações:

- 1: Cada dia de atraso acarreta uma perda de 1 ponto no exercício.
- 2: Este EP deve ser resolvido individualmente. Não serão aceitos EPs iguais ou em grupo.
- 3: A saída P.JPG não precisa ser elaborada como no exemplo. Basta ter dois pontos coloridos indicando as duas localizações do padrão buscado  $Q$ . Você pode utilizar no seu programa a informação de que cada padrão ocorre exatamente duas vezes na imagem A.
- 4: Pode usar (ou não) as funções da biblioteca Proeikon. Não utilize os programas prontos do Proeikon (chamados através de IMG NomeProg, MLE NomeProg, etc). Na biblioteca Proeikon, a seguinte função traça uma reta:

```
void reta(IMGCOR& a, int l1, int c1, int l2, int c2, COR cor);
```

A seguinte função escreve um "string":

```
void puttxt(IMGCOR& a, int la, int ca, string st,
```

```
COR fore=COR(0,0,0), int ampl=1,  
bool transp=true, COR back=COR(255,255,255));
```

- 5:** Teste o seu programa para as imagens A\*.JPG (quantidade 3) e Q\*.JPG (quantidade 12) disponíveis no site:  
**<http://www.lps.usp.br/~hae/psi5796-05/ep1-2008/imagens>**  
Nesse diretório, também tem a saída P01.JPG obtida processando a imagem A01.JPG.
- 6:** Todas as imagens foram obtidas escaneando as figuras de um jogo de memória, utilizando sempre a mesma configuração do scanner. Portanto, as figuras estão rotacionadas, mas as suas escalas são sempre iguais. O brilho/contraste também não deve mudar (de forma significativa) de uma imagem para outra.
- 7:** Programas mais rápidos receberão notas maiores. Programas que cometem erros de localização terão nota descontada.
- 8:** Se você fez os programas utilizando Devcpp/Proeikon, entregue apenas o programa fonte EP1.CPP, os três arquivos de saída P\*.JPG obtidos rodando o seu programa e um documento coment.pdf (Acrobat) ou coment.doc (MS-Word) com os comentários que achar convenientes. Neste documento, descreva (em português) o seu método, a seqüência de operações efetuadas e inclua (se quiser) as imagens intermediárias que possibilite entender o método. Informe também o tempo de processamento e o modelo do seu computador (a velocidade do clock). Se você fez os programas utilizando compilador ou biblioteca diferente de Devcpp/Proeikon, deve entregar também o programa executável (EP1.EXE). **Cuidado:** Alguns servidores de emails possuem anti-vírus que bloqueia envio/recepção de emails com arquivo EXE embutido.
- 9:** Compacte todos os arquivos como **SeuNome\_Sobrenome.ZIP** e envie um email colocando como assunto **“PSI5796 EP1”** para o endereço abaixo:
- **hae@lps.usp.br**
- 10:** Procure enviar um único email para entregar o seu EP. Se você enviar dois ou mais emails, vou apagar todos os emails antigos, considerando somente o email enviado por último.
- 11:** Vou corrigir o EP usando imagens a\*.jpg diferentes (mas obtidos da mesma forma) das imagens disponíveis no site.