

Exercício-Programa 1: Bafômetro

Apresentação:

Utilize um algoritmo **tentativa e erro** para resolver o problema descrito a seguir.

Um estudante universitário chamado Jorge foi com seus amigos a um bar, comemorar o final do semestre. Embora Jorge tenha ido de carro, consumiu uma razoável quantidade de cerveja. Nessa noite, a polícia estava nas ruas do bairro, pronta para prender quem estivesse dirigindo sob influência de álcool. Apesar do estado inebriante de sua mente, Jorge percebeu que poderia evitar a polícia enquanto dirigia para casa.

Sua estratégia para tentar chegar em casa foi a seguinte: ao chegar em uma esquina, ele olharia em direção aos cruzamentos seguintes, um de cada vez, e prosseguiria em direção à primeira esquina não visitada anteriormente e onde não houvesse sinal de polícia. Chegando a um cruzamento, se a única rua não bloqueada fosse exatamente aquela por onde havia vindo, ele retornaria por esta rua e continuaria o algoritmo a partir do cruzamento anterior.

Naquela noite a polícia cobria um percentual p das esquinas (p.ex. 40%), isto é, a probabilidade de que a polícia estivesse ocupando qualquer esquina era p , e a probabilidade de que Jorge pudesse prosseguir em direção a qualquer esquina (não visitada ainda) era $1 - p$. Após a saída de Jorge, a polícia chegou ao bar. Assim, caso ele retornasse pelo caminho utilizado até o bar, seria preso e autuado.

O problema proposto é: dada a configuração do bairro (número de quarteirões na vertical e na horizontal, coordenadas do bar e coordenadas da casa) e dada a cobertura p da polícia, dê uma estimativa da probabilidade de que Jorge seja capaz de chegar em sua casa sem ser preso, partindo do bar.

Implementação:

Parâmetros de Entrada: O programa deverá ser chamado com os seguintes parâmetros:

- $qtlin$, $qtcot$ (inteiros positivos): quantidade de linhas e colunas da matriz de esquinas
- $xBar$, $yBar$ (inteiros positivos; $0 \leq xBar < qtlin$, $0 \leq yBar < qtcot$): coordenadas do bar
- $xCasa$, $yCasa$ (inteiros positivos; $0 \leq xBar < qtlin$, $0 \leq yBar < qtcot$): coordenadas da casa do Jorge
- *semente* (inteiro longo não negativo): semente do gerador de numeros aleatorios para distribuicao probabilística da polícia entre as esquinas; se o valor for 0, a escolha da semente será automática.
- $ProbPol_1, ProbPol_2 \dots ProbPol_k$ (ponto flutuante, $0 < ProbPol_i < 1$, $i = 1, 2, \dots, k$) vetor contendo as probabilidades da policia estar presente em cada esquina.

Resultados: O programa deverá ter dois comportamentos distintos e produzirá dois resultados, conforme o número de parâmetros de cobertura informados:

1. Caso apenas uma cobertura tenha sido informada como parâmetro:
 - Sorteie a distribuição da polícia no bairro;
 - Rode o algoritmo de tentativa e erro e imprima o resultado (Sucesso/Fracasso)
 - Imprima na tela o conteúdo da matriz de esquinas. (A especificação dos códigos da matriz é detalhada no código-fonte base disponível na página da disciplina.)
2. Caso tenham sido informados dois ou mais valores de cobertura como parâmetros:
 - Para cada valor de cobertura, simular 1000 cenários, cada um consistindo em:
 - Sortear a distribuição da polícia no bairro
 - Rodar o algoritmo tentativa e erro
 - Incrementar o contador de sucessos e somar o comprimento do caminho
 - Imprimir na tela uma matriz com os resultados da simulação. Essa matriz conterá k linhas (onde k é o tamanho do vetor de coberturas) e 3 colunas, contendo:
 - Resultado[i][0]: Probabilidade de ocorrência de policiais no bairro
 - Resultado[i][1]: Proporção de tentativas bem sucedidas de chegar em casa
 - Resultado[i][2]: Comprimento médio dos caminhos bem sucedidos; 0 se não houver nenhum caminho bem sucedido para a cobertura solicitada

Exemplos:

1) A chamada do programa com os parâmetros:

```
java ep1234567 7 8 6 5 2 1 321 0.4
```

deverá gerar os seguintes resultados na tela:

```
Sucesso - caminho de tamanho 26
  0      0      0    -10     13     12     11      0
  0      0      0    -10     14      9     10    -10
-10     -2      0    -10     15      8    -10      0
  0     25    -10      0     16      7      6      5
-10     24     23      0     17    -10      3      4
-10      0     22    -10     18      1      2    -10
-10    -10     21     20     19     -1    -10    -10
```

2) A chamada do programa com os parâmetros:

```
java ep1234567 7 8 6 5 2 1 321 0.1 0.2 0.3 0.4 0.5 0.6
```

deverá gerar os seguintes resultados na tela:

```
0.100000 0.973000 36.189106
0.200000 0.865000 28.030058
0.300000 0.606000 21.198020
0.400000 0.334000 16.209581
0.500000 0.142000 13.309859
0.600000 0.037000 10.054054
```

Entrega do trabalho:

Condições da entrega:

- Deverá ser entregue um único módulo em java nomeado na forma *ep<numerosp>.java*. Este módulo conterá a implementação completa do programa.
- O código-fonte deverá ser compilável no DrJava.
- O trabalho deverá ser enviado para marcelolauretto@usp.br com o assunto ACH2002-EP1.
- O prazo para entrega é 23/05/2010.
- Dúvidas a respeito das especificações ou a respeito da implementação do trabalho serão sanadas até o dia 20/05/2010. Dúvidas encaminhadas após este prazo serão ignoradas.
- Além da correção do programa, serão consideradas a clareza e a documentação (comentários) do código.
- O trabalho será individual. Se houver evidência de plágio entre trabalhos, os mesmos serão desconsiderados.