

POO – 2.o Semestre de 2003

Aula 8 – Exercícios

Profs. Fábio de Miranda e Carlos dos Santos

Dica para todos os exercícios:

- Trabalhe no projeto do BlueJ extraído de *aula8.zip*, pois este já contém diversas das classes que vamos usar.
- Utilize intensamente os javadocs. Eles começam no arquivo *docs/api/index.html* do diretório de instalação do Java.

Problema 1 – Skodak's IO Helper

Neste problema você vai praticar uma forma simplificada de fazer leitura e escrita de arquivos, que é através do uso das classes *IOHelper*, *SimpleReader* e *SimpleWriter*. Para fazer este exercício, você deve extrair o arquivo chamado *aula8.zip* e ler a documentação das três classes mencionadas acima.

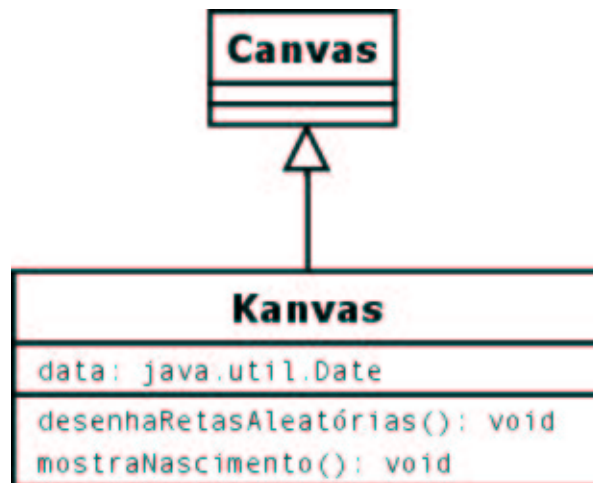
O que o exercício pede:

- Faça uma classe chamada *Capitaliza* que em seu método *main* aceite dois argumentos. O primeiro argumento tem o nome de um arquivo de entrada, e o segundo argumento um arquivo de saída. O arquivo de saída deve conter uma versão do arquivo de entrada com todos os caracteres convertidos para letras maiúsculas.

Dicas:

- Rode a partir da linha de comando para que o BlueJ não se confunda com o *path* dos arquivos. Não se esqueça de que, como sua classe agora pertence a um pacote, muda o modo de se fazer a invocação na linha de comando.
- Utilize *IOHelper* para criar *SimpleReader* e *SimpleWriter*. Não chame os construtores destas duas classes diretamente.

Problema 2 – Retas aleatórias



- a) Crie uma subclasse da classe *Canvas*, chamada *Kanvas*, que tenha um método chamado *desenhaRetasAleatórias()*. Este método deve utilizar a classe *SimpleInput* para perguntar ao usuário quantas retas (digamos n) deverão ser criadas, e depois desenhar sortear $2n$ pontos aleatórios e traçar retas ligando cada par destes pontos. Cada reta deverá ser de uma cor diferente.
- b) Adicione um atributo do tipo [java.util.Date](#) à sua classe, e o instancie quando o construtor de sua classe for chamado. Aproveitando este *Date*, crie um método chamado *mostraNascimento* que imprime no centro do *Kanvas* a data representada por *Date*.

Dicas:

- Crie somente um construtor vazio para sua classe. Isto acarretará um erro de compilação. Chame o professor quando isto acontecer se não entender a mensagem do compilador. Anote em algum lugar porque isto pode cair em provas.
- Para gerar os números aleatórios, você pode utilizar o método `random()` da classe `Math` ou então criar um objeto da classe [java.util.Random](#). Leia os Javadocs e diga qual você prefere.
- Para mudar a cor das retas, use o método `setForegroundColor(Color c)` de `Canvas`, e aproveite as cores declaradas como atributos estáticos em [java.awt.Color](#). Não é preciso criar nenhum objeto do tipo `Color`.

Problema 3

Observe a classe *Canvas*, você deverá perceber que há um método chamado *recebeEventoDeTeclado(String s)*. Instancie um *Canvas* graficamente via BlueJ, clique sobre ele e comece a teclar e a observar o que acontece no terminal. Entenda o que está acontecendo.

O que o exercício pede:

- Crie uma subclasse de *Canvas* chamada *Joguinho* que sobrecreve o método *recebeEventoDeTeclado(String s)* e se vale das teclas direcionais para controlar o movimento de algo na tela.

Dica:

Lembre-se de que há alguém (o processo controlador de interface gráfica do Java) chamando *recebeEventoDeTeclado(s)* para você. Não vai ser necessário escrever nenhum loop infinito para controlar a simulação.

Problema 4

Pense sobre o que o professor fez na classe *Canvas* para o *Problema 3* – para se aproveitar da classe *Canvas* editou seu código fonte criando uma versão diferente da criada pelo autor original. Esta foi a melhor maneira de se basear na funcionalidade da classe *Canvas*? Responda num comentário dentro do código fonte de sua classe *Joguinho*.