

# Tarea No 1

Prof.: Kurt Schwarze / Entrega Martes anterior al control 1

15th April 2004

## 1 Dados

Considera la clase Dado

Método	Descripción
new Dado(int n)	construye un dado de n caras
void tirar()	tira el dado
int valor()	devuelve el valor del dado

Escribe un programa que calcule en forma numérica la probabilidad de que al tirar 3 dados de 7 caras, los dados aparezcan ordenados (es decir,  $\text{dado1} > \text{dado2} > \text{dado3}$  o  $\text{dado1} < \text{dado2} < \text{dado3}$ ).

Para la simulación use 1.000.000 de tiros. La salida del programa debe ser

```
La probabilidad de que los dados salgan  
ordenados es de: 15.8%
```

Nótese que la probabilidad tiene un solo decimal.

**Nota** no use `Math.random()`. La clase `Dado` puede ser descargada de `ucursos`.

## 2 Severla

El ranking top-top-top-top-top, incluye el hit Severlá. Fuentes de inteligencia sospechan que la letra de la canción contiene algún tipo de mensaje subliminal, aunque ignoran que pueda ser. Escribe un programa que lea el archivo `SEVERLA.TXT` y lo escriba decodificado en `ALREVES.TXT`.

**Nota** Debes crear un método con la siguiente firma

```
String desenreda(String s)
```

Este método recibe una línea del archivo y la devuelve descryptada. La letra de la canción puede ser obtenida mediante Google buscando “severla otto”.

### 3 Mezclas

Guillermo T., el tabernero, destila sus propios tragos (bautizados con números: 1,2,3,4,5) . Después de ciertos experimentos con sus clientela, ha descubierto que una curiosa reacción química hace que las personas exploten cuando toman el trago n seguido del trago n+1 ( $0 < n < 5$ ). Para evitarlo, le pide que le desarrolle una pequeña aplicación que sostenga el siguiente diálogo:

```
Ingrese el primer trago: 2
Ingrese el siguiente trago: 4
Puede tomarlo
Ingrese el siguiente trago: 5
No puede tomarlo
Ingrese el siguiente trago: 5
No puede tomarlo
Ingrese el siguiente trago: 1
Puede tomarlo
Ingrese el siguiente trago: 0
Gracias por usar el programa
```

### 4 Punto

Implementa la clase Punto, con los siguientes métodos

Método	Descripción
new Punto()	Construye un punto en el origen
new Punto(double x, double y)	Construye un punto (x,y)
String toString()	Devuelve una representación de la forma (x,y)
Punto opuesto()	Devuelve un nuevo Punto, opuesto por según el origen
double distancia()	Retorna la distancia al origen
double distancia(Punto p)	Devuelve la distancia al punto p
void setX(double x)	Fija la coordenada X
void setY(double y)	Fija la coordenada Y

#### 4.1 Representación x,y

Escribe la clase Punto usando como representación dos doubles, una para x, el otro para y

#### 4.2 Representación con $r, \theta$

Nuevamente lo mismo, pero con r y theta como representación interna

#### 4.3 Uso

Escribe un programa que use todos los métodos de la clase Punto