
Práctica 3

Fecha de entrega: 20/21 de Abril

OBJETIVO: Practicar con enumerados, registros y vectores.

Esta tercera práctica consiste en la creación de dos pequeñas aplicaciones: la primera simula el movimiento de una bola en un *pinball* y la segunda simula los movimientos de un caballo sobre un tablero de ajedrez.

1. Primera Parte

Un juego de Pinball tiene una bola que se mueve por una matriz de $N \times M$ casillas, numeradas de $1..N$ filas x $1..M$ columnas.

La matriz está formada por *casillas en blanco* por las que se puede mover la pelota (y que serán representadas por un carácter en blanco), y *casillas obstáculo* en las que la bola rebota (y que serán representadas por algún carácter parecido a un bloque, por ejemplo el código ASCII 178).

El movimiento de la bola sólo se realiza según alguna de las seis direcciones cardinales, Norte, Sur, Este, Oeste, Noreste, Noroeste, Sureste y Suroeste. La bola mantiene la dirección del movimiento hasta encontrar un obstáculo o el borde de la matriz (o tablero).

Cuando el movimiento de la bola se ve interrumpido por un obstáculo (o por el borde de la matriz), ésta *rebota aleatoriamente* en cualquiera de las seis direcciones cardinales indicadas anteriormente, siempre y cuando pueda hacerlo (siempre podrá rebotar en alguna dirección).

En la figura 1 se ve que la bola pasa por las siguientes situaciones: $\langle (1, 1), SE \rangle$, $\langle (2, 2), SE \rangle$, $\langle (1, 3), NE \rangle$, $\langle (1, 4), E \rangle$, $\langle (1, 5), E \rangle$, $\langle (2, 5), S \rangle$, . . .

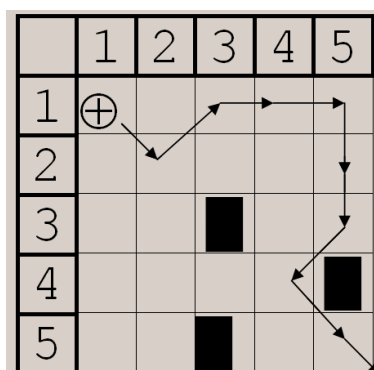


Figura 1: Ejemplo de movimientos de la bola

Se pide implementar un programa **Pr3A.pas** que simule el movimiento de la bola en un pinball de 8 filas y 5 columnas. La bola sale de la esquina superior izquierda con dirección SE. El programa mostrará por pantalla todo el recorrido de la bola por la tabla. El juego finaliza cuando la bola salga por la esquina inferior derecha. El usuario puede seleccionar el número de obstáculos que se colocarán aleatoriamente.

Para calcular aleatoriamente una nueva dirección cardinal de movimiento en caso de encontrar un obstáculo se calculará una dirección aleatoria correcta en la que se podrá continuar moviendo la bola. Es importante que la situación refleje la dirección que trae la bola, para seguirla en caso de que no haya rebote. Añade un reloj digital que muestre el tiempo que va transcurriendo (formato HH:MM:SS).

Funciones y procedimientos de Turbo Pascal: en esta práctica pueden utilizarse, entre otros, los siguientes subprogramas estándar de Turbo Pascal: **clrscr**, **gotoXY**, **delay**, **keyPressed** y **readKey**, todos ellos de la unidad CRT.

2. Segunda parte

Escribe un programa **Pr3B.pas** que simule los movimientos del caballo en el tablero de ajedrez, de modo que partiendo de la casilla 1,1, intente recorrer el máximo número de casillas del tablero sin repetir ninguna posición. En cada casilla se grabará el número del salto dado por el caballo cuando ocupó la casilla, hasta rellenar el tablero o quedarse sin poderse mover.

El movimiento del caballo consiste en desplazarse dos casillas en horizontal y una en vertical o dos casillas en vertical y una en horizontal.

El juego tendrá tres modalidades:

1. Se intentará hacer cada uno de los 8 posibles movimientos del caballo siguiendo un orden hasta que se pueda realizar uno.

2. Cada posible movimiento del caballo se hará en una dirección aleatoria utilizando la función `random(N)`.
3. Salen dos caballos de las casillas 1,1 y N,N respectivamente, y mediante la técnica del apartado 2), ganará el caballo que más casillas cubra.

3. Implementación

- Tanto el pinball como el tablero se representarán mediante estructuras de dos dimensiones.
- Se utilizarán tipos enumerados para las diferentes direcciones (*tDireccion*).
- Para la primera parte se recomienda utilizar una estructura tipo registro (*tSituacion*) para guardar la situación actual de la bola (dirección que lleva la bola y coordenadas X,Y).
- Se deben utilizar *constantes* para definir el tamaño del tablero.

4. Normas de entrega

La práctica debe entregarse utilizando el mecanismo de entregas del campus virtual, no más tarde de la fecha indicada en el encabezado del enunciado.

Sólo uno de los dos miembros del grupo debe hacerlo, subiendo al campus un fichero llamado **grupoNN.zip**, donde NN representa el número de grupo con dos dígitos. El fichero contendrá una carpeta con nombre **grupoNN** en la que se incluirá código completo de la práctica, con los ficheros **Pr3A.pas** y **Pr3B.pas**. Adicionalmente deberá incluir un fichero **alumnos.txt** donde se indique el nombre de los componentes del grupo.

No olvidar incluir las directivas de compilación en el código fuente de todos los ficheros.