PRÁCTICA 8

# // EJ 1 (HECHO)

**program {**

**IrAlBorde\_(Oeste)**

**}**

**function estoyEnElBorde(dirección) {**

**/\***

**PROP.: Indica si el cabezal esta en el borde al \*direccion\***

**PREC.: Ninguna**

**\*/**

**return (puedeMover(dirección))**

**}**

**procedure IrAlBorde\_(dirección) {**

**/\***

**PROP.: Mueve el cabezal al borde en la direccion dada**

**PREC.: Ninguna**

**\*/**

**while (not(estoyEnElBorde)) {**

**Mover(dirección)**

**}**

**}**

# **// EJ 2** (HECHO)

**procedure SacarTodasLasDeColor\_(color) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* en la celda actual**

**Prec.: Ninguna**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*/**

**while(hayBolitas(color)) {**

**Sacar(color)**

**}**

**}**

# **// EJ 3** (HECHO)

**procedure VaciarFilaDe\_(color) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* de la fila actual.**

**El cabezal puede terminar en cualquier celda**

**Prec.: Ninguna**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*/**

**IrAlBorde(Oeste)**

**while(puedeMover(Este)) {**

**SacarTodasLasDeColor\_(color)**

**Mover(Este)**

**}**

**SacarTodasLasDeColor\_(color)**

**}**

**program {**

**VaciarFilaDe\_(Rojo)**

**}**

# **// EJ 4** (HECHO)

**procedure VaciarFilaDe\_HaciaEl\_(color, dirección) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* de la fila actual.**

**El cabezal puede terminar en cualquier celda**

**Prec.: El parametro \*dirección\* solo puede valer Este u Oeste**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*dirección\* - Dirección: Indica la dirección a moverse**

**\*/**

**IrAlBorde(opuesto(dirección))**

**while(hayBolitas(color) && puedeMover(dirección)) {**

**SacarTodasLasDeColor\_(color)**

**Mover(dirección)**

**}**

**SacarTodasLasDeColor\_(color)**

**}**

**procedure SacarTodasLasDeColor\_(color) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* en la celda actual**

**Prec.: Ninguna**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*/**

**while(hayBolitas(color)) {**

**Sacar(color)**

**}**

**}**

**program {**

**VaciarFilaDe\_HaciaEl\_(Rojo, Este)**

**}**

# **// EJ 5 (**HECHO**)**

**procedure VaciarTableroDe\_(color) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* del tablero.**

**El cabezal puede terminar en cualquier celda.**

**Prec.: Ninguna**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*/**

**while(puedeMover(Norte)) {**

**VaciarFilaDe\_HaciaEl\_(color, Este)**

**Mover(Norte)**

**}**

**VaciarFilaDe\_HaciaEl\_(color, Este)**

**}**

**procedure VaciarFilaDe\_HaciaEl\_(color, dirección) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* de la fila actual.**

**El cabezal puede terminar en cualquier celda**

**Prec.: El parametro \*dirección\* solo puede valer Este u Oeste**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*dirección\* - Dirección: Indica la dirección a moverse**

**\*/**

**IrAlBorde(opuesto(dirección))**

**while(hayBolitas(color) && puedeMover(dirección)) {**

**SacarTodasLasDeColor\_(color)**

**Mover(dirección)**

**}**

**SacarTodasLasDeColor\_(color)**

**}**

**procedure SacarTodasLasDeColor\_(color) {**

**/\***

**Prop.: Quitar todas las bolitas del color \*color\* en la celda actual**

**Prec.: Ninguna**

**Param: \*color\* - Color: Indica el color de las bolitas a sacar**

**\*/**

**while(hayBolitas(color)) {**

**Sacar(color)**

**}**

**}**

**program {**

**VaciarTableroDe\_(Rojo)**

**}**

# **// EJ 6** (HECHO)

## BIBLIOTECA

### **A)**

**procedure IrAPrimeraCeldaEnUnRecorridoAl\_Y\_(dirPrincipal, dirSecundaria){**

**/\***

**Propósito: Mueve el cabezal a la primer celda en un recorrido en dirección \*\*dirPrincipal\*\* y**

**\*\*dirSecundaria\*\*.**

**Parámetros:**

**\* dirPrincipal: Dirección - La dirección principal del recorrido.**

**\* dirSecundaria: Dirección - La dirección secundaria del recorrido.**

**Precondiciones:**

**\* Las direcciones no pueden ser opuestas.**

**\* Las direcciones no pueden ser iguales.**

**\*/**

**IrAlBorde(opuesto(dirPrincipal))**

**IrAlBorde(opuesto(dirSecundaria))**

**}**

### **B)**

**function haySiguienteCeldaEnUnRecorridoAl\_Y\_(dirPrincipal, dirSecundaria){**

**/\***

**Propósito: Indica si hay una siguiente celda en un recorrido al \*\*dirPrincipal\*\* y**

**\*\*dirSecundaria\*\*.**

**Parámetros:**

**\* dirPrincipal: Dirección - La dirección principal del recorrido.**

**\* dirSecundaria: Dirección - La dirección secundaria del recorrido.**

**Precondiciones:**

**\* Las direcciones no pueden ser opuestas.**

**\* Las direcciones no pueden ser iguales.**

**Tipo: Booleano.**

**\*/**

**return(puedeMover(dirPrincipal) || puedeMover(dirSecundaria))**

**}**

### C)

**procedure IrASiguienteCeldaEnUnRecorridoAl\_Y\_(dirPrincipal, dirSecundaria){**

**/\***

**Propósito: Mueve el cabezal a la siguiente celda en un recorrido en dirección \*\*dirPrincipal\*\* y \*\*dirSecundaria\*\*.**

**Parámetros:**

**\* dirPrincipal: Dirección - La dirección principal del recorrido.**

**\* dirSecundaria: Dirección - La dirección secundaria del recorrido.**

**Precondiciones:**

**\* Las direcciones no pueden ser opuestas.**

**\* Las direcciones no pueden ser iguales.**

**\*/**

**if(puedeMover(dirSecundaria)) {**

**Mover(dirSecundaria)**

**}**

**else{**

**IrAlBorde(opuesto(dirSecundaria))**

**Mover(dirPrincipal)**

**}**

**}**

# // EJ 7

# // EJ 8

# // EJ 9

# // EJ 10

## 

# // EJ 11

# // EJ 12

# // EJ 13

# // EJ 14