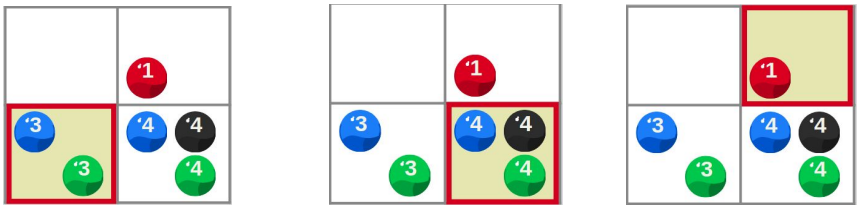
Introducción a la Programación - Práctica 3
Alternativa condicional y funciones

1) Indicar el valor y el tipo que representan las siguientes expresiones en cada uno de los Tableros A , B y C .



(A) (B) (C)

1. `not hayBolitas(Rojo)`
2. `puedeMover(Norte) && puedeMover(Este)`
3. `puedeMover(Norte) || puedeMover(Este)`
4. `not puedeMover(Norte) && puedeMover(Este)`
5. `nroBolitas(Negro) + nroBolitas(Azul)`
6. `nroBolitas(Negro) == nroBolitas(Azul) && nroBolitas(Negro) == nroBolitas(Verde)`
7. `puedeMover(opuesto(opuesto(Este)))`

2) Escribir expresiones que denoten

1. **`True`** cuando la celda actual tiene más de 5 bolitas en total.
2. **`True`** cuando la celda actual tiene al menos 5 bolitas en total.
3. **`True`** cuando la celda actual tiene al menos 5 bolitas en total y el borde se encuentra justo al este de la misma.
4. **`True`** cuando la celda actual tiene una celda vecina al **`Norte`** o al **`Este`**.
5. **`True`** cuando la celda actual tiene bolitas de todos colores.
6. **`True`** cuando en la celda actual faltan bolitas de al menos un color (dar una solución sin usar la función del ítem anterior y otra usándola).
7. **`True`** cuando hay una celda al **`Este`** con bolitas negras o rojas (no hay precondición).

3) Escribir un procedimiento **`SacarSiHay(color)`** que, dado un color **`color`**, saque una bolita de color **`color`** de la celda actual. Si no hubiera bolitas de color **`color`** el procedimiento no hace nada.

4) Escribir el procedimietno PonerVerdeSiHayNegroYAzul(), que pone una bolita de color Verde si hay una bolita de color Negro y una bolita de Color Azul en la celda actual

5) Escribir el procedimiento **`PonerSoloSiHayAmbasAMirar(colorAPoner, colorAMirar1, colorAMirar2)`** que agregue una bolita de color **`colorAPoner`** a la celda actual si hay una bolita de color **`colorAMirar1`** y otra de color **`colorAMirar2`** en la misma.

6) Escribir un procedimiento **`DesempatarDosColores(color1,color2)`** que, dados dos colores **`color1`** y **`color2`**, ponga una bolita de color **`color1`** si la celda actual contiene la misma cantidad de bolitas de color **`color1`** y **`color2`**.

7) Escribir un procedimiento **PonerSiMásSiHayVerde()**, que pone una bolita de color Verde si ya hay alguna bolita de color Verde en la celda actual

8) Escribir un procedimiento **PonerFlechaNorteSiHayLugar(flechaNorte)**, que dado un color que representa la flecha al Norte, dibuje una flecha al Norte si existe espacio para moverse en esa dirección. Las flechas al Norte será representada con una bolita de color flechaNorte. Recordá no mezclar los niveles de abstracción del problema y definir otros procedimientos para eso si fuera necesario.

9) Escribir un procedimiento **`PonerSi`** que dado un color **`color`** y un booleano **`bool`**, ponga una bolita de color **`color`** en la celda actual si **`bool`** es **`True`**, y no haga nada si este es **`False`**.

10) Utilizando el ejercicio anterior, escriba los procedimientos **`PonerSoloSiHayAmbasAMirar`** y **`DesempatarDosColores`**. ¿Que beneficios trae tener un procedimiento **`PonerSi`** contra utilizar **`if`** en cada caso?

11) Escriba los procedimientos **`SacarSi(color, bool)`** y **`MoverSi(dirección,bool)`** que actúan de forma similar a **`PonerSi`**.

12) Considere el problema de sacar 8 bolitas de color **`Azul`**. La precondición, como es de esperar, es que haya al menos 8 bolitas azules en la celda actual. Sólo uno de los siguientes procedimientos resuelve correctamente dicho problema. ¿Cuál es? Justifique.

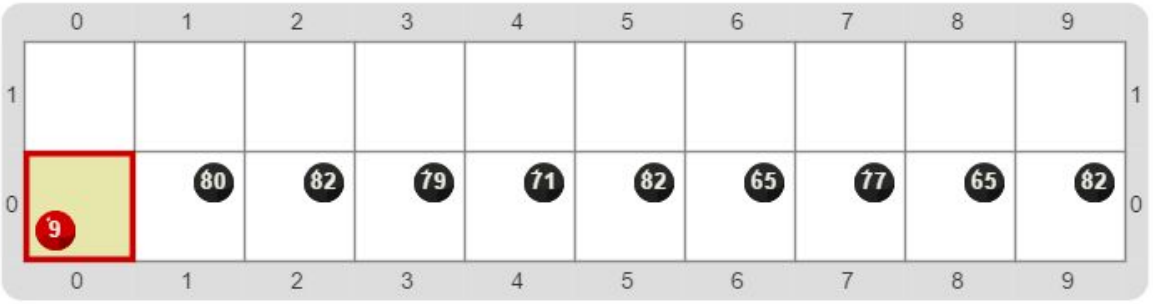
```
procedure QuitarOchoBolitasA () {
      repeat (8) {
            Sacar(Azul)
      }
}

procedure QuitarOchoBolitasB () {
      repeat (8) {
            SacarSiHay(Azul)
      }
}
```

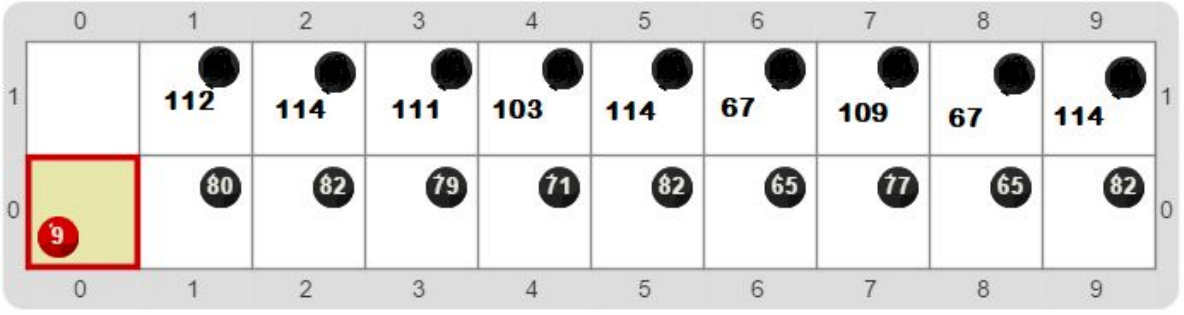
13) Hacer el procedimiento **`PasarAMinúscula()`** que asumiendo que en la celda actual se codifica con bolitas rojas la cantidad de letras representada en la fila actual empezando en la celda al este, ponga la misma palabra en mayúsculas en la fila al Norte.

-Las letras se representan con bolitas negras.
- Se sabe que en el código ASCII si las letras mayúsculas se codifican con un número N entonces la misma letra minúscula se representa con N+32.

Ejemplo de Tablero inicial:



Ejemplo de Tablero Final:



14) Sobre el ejercicio de soporte técnico presentado el jueves pasado en la práctica:
- Modifique su solución, ¿donde cree útil que se pueden agregar funciones? ¿Qué ventajas ve?
- Modifique su solución nuevamente, teniendo en cuenta que ahora el procedimiento **`RepararMaquina()`**, ya NO tiene como precondición que haya virus en la máquina actual, y no debemos pasar un antivirus sobre una máquina que ya tiene la marca de Ok. Con este cambio, ¿dejaría el mismo nombre al procedimiento?

15) Suponiendo que estamos programando un juego donde en las celdas del tablero se representan Soldados (con Bolitas de color Negro), y Terroristas (con bolitas de color Rojo), escribir las siguientes funciones (defina también su contrato):

a) **`cantidadDeSoldados()`**, que devuelve la cantidad de soldados de la celda actual.
b) **`cantidadDeTerroristas()`**, que devuelve la cantidad de terroristas de la celda actual.
c) **`haySuficientesSoldados(cantidadDeSoldados)`** que devuelve True, si por lo menos hay n soldados, por cada terrorista de la celda actual
d) **`esCeldaIndefensa()`** que Devuelve True solo si no hay soldados en la celda actual.
e) **`estadoDeEmergencia()`** que devuelve True sólo si existen mas de 100 terroristas, y además la celda está indefensa
f) **`soldadosNecesariosParaDefensaEficaz(cantidadDeSoldados)`** que devuelve el número de soldados que faltan para defender la celda actual, si por cada terrorista se necesitan n soldados para combatirlos.(tener en cuenta que en la celda actual pueden existir soldados)