

Especificaciones de QDraw

Primitivas	Parcial/Total	Efecto
PintarNegro	Total	Pinta la celda actual de color negro(si había otro color en esta misma, ese color se pierde).
PintarRojo	Total	Pinta la celda actual de color rojo(si había otro color en esta misma, ese color se pierde).
PintarVerde	Total	Pinta la celda actual de color verde(si había otro color en esta misma, ese color se pierde).
Limpiar	Parcial	Remueve el color de la celda en la que se encuentra actualmente el cabezal. Esto puede fallar si no hay un color en la celda actual.
MoverArriba	Parcial	Mueve el autómeta/cabezal a la celda que está ubicada arriba de la celda actual(puede fallar si las dimensiones del tablero están acotadas).
MoverAbajo	Parcial	Mueve el autómeta/cabezal a la celda que está ubicada abajo de la celda actual(puede fallar si las dimensiones del tablero están acotadas).
MoverDerecha	Parcial	Mueve el autómeta/cabezal a la celda que está ubicada a la derecha de la celda actual(puede fallar si las dimensiones del tablero están acotadas).
MoverIzquierda	Parcial	Mueve el autómeta/cabezal a la celda que está ubicada a la izquierda de la celda actual(puede fallar si las dimensiones del tablero están acotadas).
estaPintadaDeNegro?	Total	Retorna verdadero si la celda actual está pintada de negro, si no retorna falso.
estaPintadaDeVerde?	Total	Retorna verdadero si la celda actual está pintada de verde, si no retorna falso.
estaPintadaDeRojo?	Total	Retorna verdadero si la celda actual está pintada de Rojo, si no retorna falso.
estaVacía?	Total	Retorna verdadero si la celda actual no está pintada, en caso contrario retorna falso.

CONECTORES LOGICOS

AND	V	F
V	V	F
F	F	F

OR	V	F
V	V	V
F	V	F

NOT	
V	F
F	V

Especificaciones de QDraw

Estructuras de código válidas

```
Programa {  
    ProcedimientoA()  
    ProcedimientoR()  
    ProcedimientoC()  
    ProcedimientoD()  
}
```

```
Procedimiento ProcedimientoA(){  
    /*  
    *    Documentación  
    */  
  
    BLOQUE DE CÓDIGO  
}
```

```
Procedimiento ProcedimientoR(){  
    /*  
    *    Documentación  
    */  
  
    repetir (nro) veces{  
        BLOQUE DE CÓDIGO  
    }  
}
```

```
Procedimiento ProcedimientoC(){  
    /*  
    *    Documentación  
    */  
    si ( condición ) entonces {  
        BLOQUE DE CÓDIGO  
    }  
    sino {  
        BLOQUE DE CÓDIGO  
    }  
}
```

```
Procedimiento ProcedimientoD(){  
    /*  
    *    Documentación  
    */  
    si (condición) entonces{  
        BLOQUE DE CÓDIGO  
    }  
}
```

Estructuras de código inválidas

```
Programa {  
    Procedimiento ProcedimientoA(){  
        BLOQUE DE CÓDIGO  
    }  
}
```

```
Procedimiento ProcedimientoR(){  
    /*  
    *    Documentación  
    */  
    repetir (nro) veces {  
        repetir (nro) veces {  
            BLOQUE DE CÓDIGO  
        }  
    }  
}
```

```
Procedimiento ProcedimientoC(){  
    /*  
    *    Documentación  
    */  
    si ( condición ) entonces {  
        BLOQUE DE CÓDIGO  
    }  
    sino {  
        si ( condición ) entonces {  
            BLOQUE DE CÓDIGO  
        }  
    }  
}
```

```
Procedimiento ProcedimientoD(){  
    /*  
    *    Documentación  
    */  
    Repetir (nro) veces {  
        BLOQUE DE CÓDIGO  
    }  
    Repetir (nro) veces {  
        BLOQUE DE CÓDIGO  
    }  
}
```