

ACTIVIDAD 6

PELOTA REBOTANDO

SIMULACIÓN DE MOVIMIENTO

En la actividad 5 se simulaba el movimiento de un círculo utilizando un bucle for/next para modificar la componente x (y) de la posición del círculo. Eso sólo es útil si sabemos el número exacto de repeticiones necesario para llegar al límite como ocurre en el desplazamiento horizontal (vertical) dentro de un rectángulo.

En esta actividad vamos a ver otra forma de hacerlo. Comenzaremos asignando un valor al desplazamiento horizontal y fijando la posición inicial del círculo. Para repetir el desplazamiento utilizaremos un bucle do/until.

Ejemplo: Desplazar de izquierda a derecha un círculo rojo de radio 25 píxeles sobre un rectángulo gris de 300 píxeles de ancho con incrementos de 5 píxeles

```
des=5
x=0

do
  x = x + des
  color grey
  rect 0, 0, 300, 200
  color red
  circle x, 100, 25
until x>=300
```

Si el desplazamiento queremos que sea de derecha a izquierda haremos lo mismo pero asignando un valor negativo al desplazamiento

Si el desplazamiento queremos que sea vertical haremos variar la coordenada y.

Si queremos que desplazamiento sea oblícuo haremos variar las dos coordenadas x e y a la vez.

Para que sea un movimiento continuo hay que detectar cuando se alcanzan los límites del rectángulo e invertir el sentido del movimiento cambiando el signo al desplazamiento.

EVITAR EL PARPADEO

Al representar movimientos es frecuente que la imagen presente un parpadeo que puede ser muy molesto. Para evitarlo se utilizan las instrucciones [fastgraphics](#) y [refresh](#). Al declarar fastgraphics las instrucciones gráficas no se representan en la pantalla hasta que se utiliza la instrucción refresh, en ese momento se actualiza la pantalla gráfica presentando todos los gráficos a la vez. Si no se utiliza la instrucción refresh los gráficos no se representan, por lo que hay que acordarse de su uso.

Código del programa de la actividad 6

```
CLG
fastgraphics
Graphsize 500,500
color green
rect 0,0,500,500
color red
font "Arial",20,70
text 160, 15, "ACTIVIDAD 6"
color black
font "Arial",12,70
text 175, 465, "Nombre y Apellido"

sumax = 2  #desplazamiento horizontal
sumay = 4  #desplazamiento vertical
x=250
y=250

do
  color yellow
  rect 50,50,400,400
  x=x+sumax
  if x>=480 then
    x=480
    sumax= -sumax
  endif
  if x<=20 then
    x=20
    sumax= -sumax
  endif
  y=y+sumay
  if y>=480 then
    y=480
    sumay= -sumay
  endif
  if y<=20 then
    y=20
    sumay= -sumay
  endif
  color blue
  circle x,y, 20
  refresh
  pause 0.01
until key>0
```

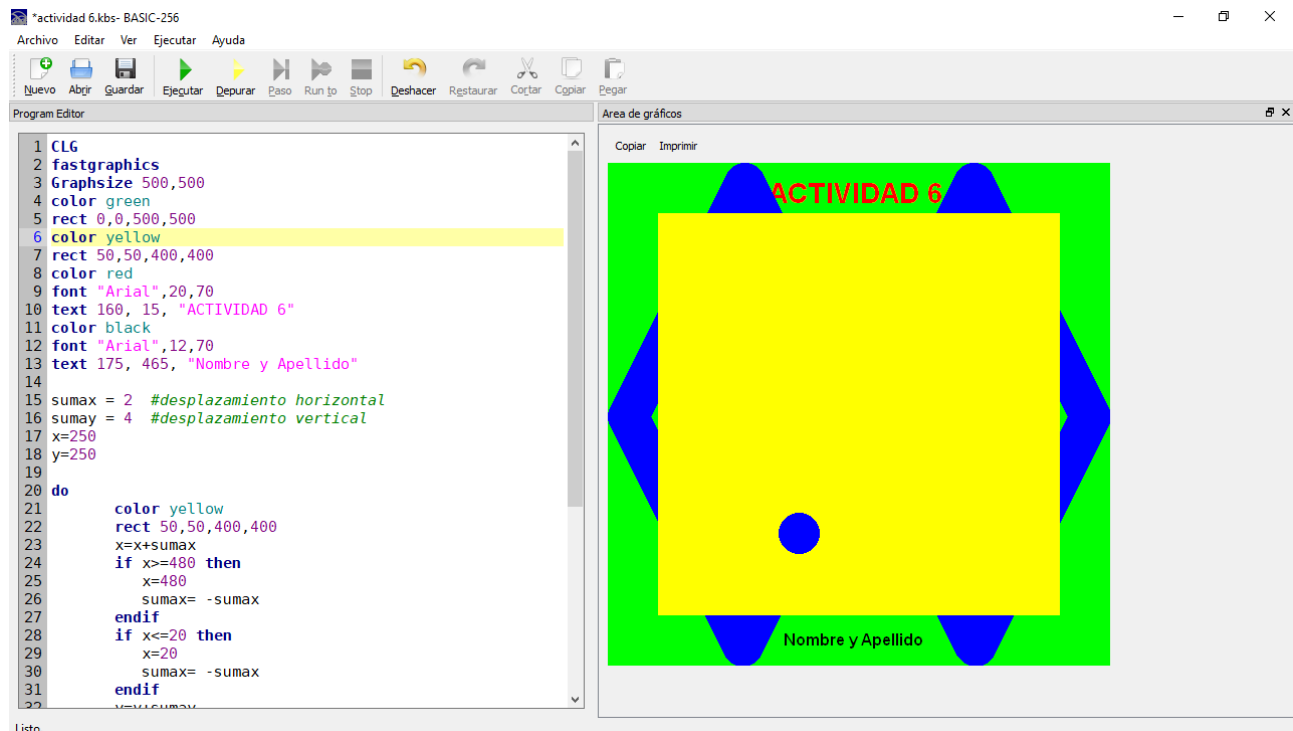
ACTIVIDAD

(Si no has podido instalar el programa, inténtalo hacer sobre el texto)

Copia en Basic256 el código que aparece sobre estas líneas, guárdalo como actividad 6, ejecútalo para comprobar si funciona bien y completa la actividad con:

1. Modifica la segunda línea de texto para que escriba tu nombre y apellido
2. Modifica el color del círculo
3. Haz los cambios que consideres necesarios para que el círculo no salga del rectángulo amarillo
4. Vuelve a guardar el programa
5. Elimina las instrucciones dentro del bucle do/until que asignan color y dibujan el rectángulo amarillo.
6. Ejecuta varias veces el programa modificando los valores de sumax y sumay: 5 y 3, 3 y 4, 6 y 2
7. Haz una captura de pantalla de una de las tres opciones del punto 6
8. Envíala por correo electrónico

Al ejecutar inicialmente el programa quedará como sigue:



Al ejecutar el programa modificado quedará algo así:

