

# Concurso interno de programación

**CIP 2010**

**FCHE**

Para el concurso se debe entregar solo el archivo fuente: nombre.c o nombre.cpp, por mail a [profesora\\_elizabeth@yahoo.com.mx](mailto:profesora_elizabeth@yahoo.com.mx); Recibiendo hasta el día **3 noviembre 2010**. El tema esta abajo.

Quienes ganen los tres primeros lugares tendrán 10 en la materia, excepto que reprobren el laboratorio. Los alumnos de laboratorio de computación para ingenieros podrán no asistir las últimas dos semanas de lab, si están entre los tres primeros lugares. Si existen programas iguales solo se tomara el primero que llegue al mail, y el otro es descartado (y tampoco contara como proyecto final). LA PARTICIPACION ES INDIVIDUAL. Deberá contener nombre completo y grupo.

**Proyecto final 2011-1:** Es el mismo tema pero puede entregarse hasta el día 18 de noviembre 2010, en salón de clase o laboratorio. LA PARTICIPACION ES INDIVIDUAL. NO PUEDE HABER PROGRAMAS REPETIDOS. SOLO EL PRIMERO QUE YO VEA CUENTA Y LAS REPLICAS SE ANULAN.

## TEMA Y REGLAS:

\* Lenguaje. C únicamente.

\*Todos utilizaran DEV C con OPENGL básico y EXCLUSIVAMENTE glBegin(GL\_POINTS) o glBegin(GL\_LINES). Checa el video.

[http://www.dailymotion.com/video/xf5wol\\_devc-con-graficas-en-opengl-basico\\_tech](http://www.dailymotion.com/video/xf5wol_devc-con-graficas-en-opengl-basico_tech)

Informe de opengl, con glBegin(GL\_POINTS) y líneas.

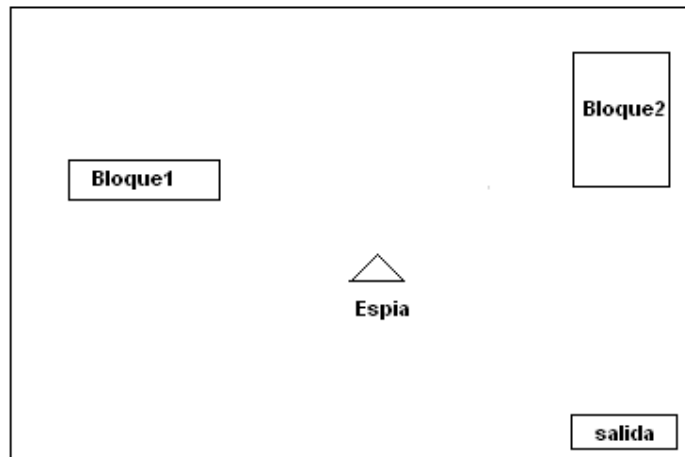
[http://sabia.tic.udc.es/gc/Tutorial%20OpenGL/tutorial/cap3.htm#\\_Toc535127328](http://sabia.tic.udc.es/gc/Tutorial%20OpenGL/tutorial/cap3.htm#_Toc535127328)

## JUEGO MAESTRO: El espía vendado. (Registrado2008-458-5468)

### INTRODUCCION:

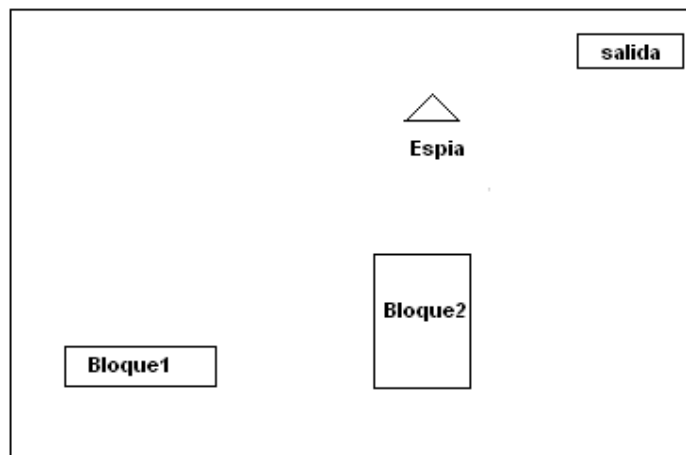
El juego empieza con la aparición repentina del *espía vendado* (*capturado por talibanes*) en alguna parte de la habitación (posición aleatoria), el espía atado y vendado de los ojos comienza a caminar porque escucha el clock de un reloj, intuye que es una bomba, toca algunas paredes, pero unas son falsas, el tiempo de reloj se comienza a acabar, si toca salida se salva, pero si el tiempo lo alcanza adentro, explota junto con todo.

#### IIICIO 1 Espia Vendado



Tiempo de Bomba: 0:30 seg.

#### IIICIO 2 Espia Vendado



Tiempo de Bomba: 0: 60 seg.

### Restricciones.

Los **bloques y la salida** tendrán una posición aleatoria y diferente en cada aparición del espía o nuevo juego.

La posición inicial del **espía** es aleatoria también por cada nuevo juego.

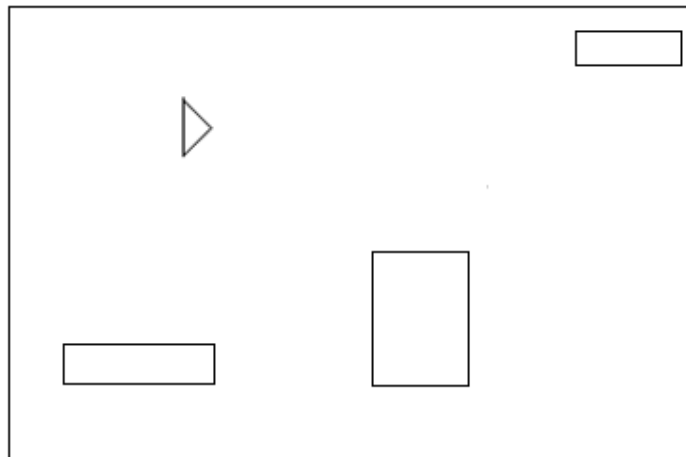
El tiempo de la **bomba** es un conteo regresivo que es aleatorio entre 30 y 60 segundos.

La talla de cada objeto debe ser muy aproximada a lo que se ve en las figuras.

El tipo de objeto y formato NO DEBE CAMBIARSE.

Solo debe aparecer el titulo del juego en la ventana, pero no se requiere para la visualización interna de letras. El tiempo de la bomba puede omitirse visualmente, pero en el proceso interno.

**Espía Vendado**



El espía (el triángulo) puede desplazarse solo a la izquierda, derecha, hacia arriba y hacia abajo.



El jugador **SOLO PUEDE ANTICIPAR SUS MOVIMIENTOS Y CHECAR SI TOCA PARED.**

En el juego se le pedirá al jugador programar a futuro los pasos y luego se corre el programa automáticamente.

En el juego se pide: (Ejemplo)

<p style="text-align: center;"><b>JUEGO del ESPIA VENDADO.</b></p> <p><b>Ordena decisiones a futuro.</b></p> <p><b>Derecho (D). Dar vuelta a la izquierda (VZ). Dar vuelta a la derecha (VD) ir para Atrás (A).</b></p> <p><b>Numero de Pasos : un paso (N1), dos pasos (N2), o tres pasos (N3)</b></p> <p><b>¿Si hay obstáculo? Numero de ordenes: <u>3</u></b></p> <p>1: <u>VZ</u></p> <p>2: <u>D</u></p> <p>3: <u>N3</u></p>
---

Lo que esta subrayado es el usuario que respondió al programa.

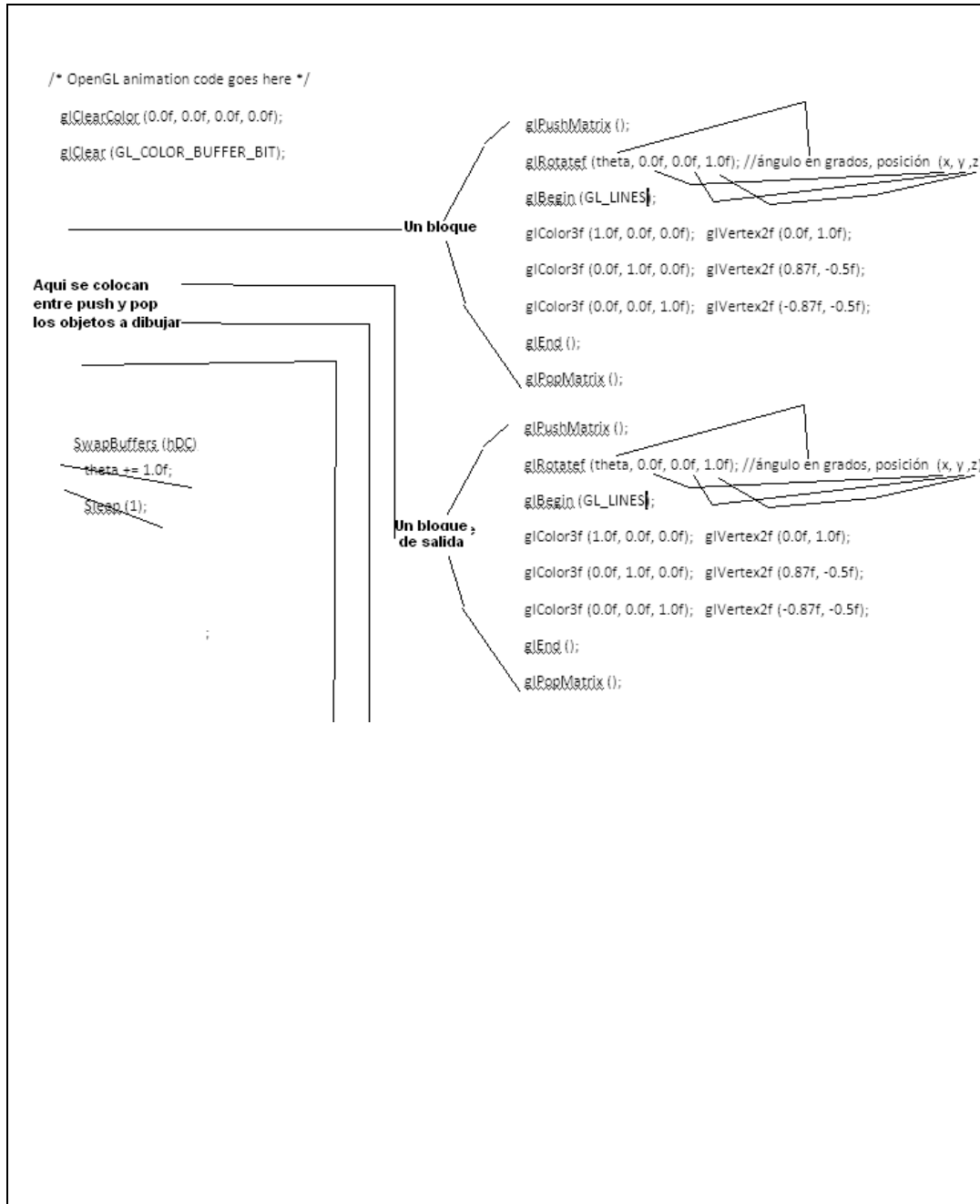
<p style="text-align: center;"><b>JUEGO del ESPIA VENDADO.</b></p> <p><b>Ordena las decisiones a futuro.</b></p> <p><b>Derecho (D). Dar vuelta a la izquierda (VZ). Dar vuelta a la derecha (VD) ir para Atrás (A).</b></p> <p><b>Numero de Pasos : un paso (N1), dos pasos (N2), o tres pasos (N3)</b></p> <p><b>¿Si No hay obstáculos? Numero de ordenes: <u>3</u></b></p> <p>1: <u>D</u></p> <p>2: <u>N3</u></p> <p>3: <u>N3</u></p>
---

El jugador solo deberá seleccionar que desea hacer para organizar sus órdenes y al terminar de escribirlas el juego comienza, todo AUTOMATICAMENTE. El triangulo comienza a moverse, el tiempo comienza a pasar, los obstáculos y la salida la vemos en una posición dada, el espía aparece en algún lugar dentro de la habitación (rectángulo). Una buena estrategia lo salvara.

El programa se deberá realizarse únicamente con líneas o puntos de opengl y glBegin.

Los datos aleatorios pueden generarse con rand();

El movimiento del espía puede llevarse con glTranslatef(x,y,z) a la posición siguiente.



glRotatef(ángulo en grados, posx, posy, posz), con un ángulo que gira en torno a un eje

si es x= 1,0,0, si es eje y=0,1,0 o eje z=0,0,1. Para rotar al espía.

Los vértices glVertex2f(1.0,2.0), glVertex3f(1.1, 0.5,1.5) son puntos de donde se imprime un punto o se une una línea. Para hacer cada objeto: La habitación: un rectángulo; dos bloques (dos rectángulos), un bloque de salida(otro rectángulo), y un triangulo(el espía)

Para colorear los vértices antes se ponen `glColor3f(rojo, verde, azul)`

Si se desea rojo se pone 1,0,0, si se desea verde 0,1,0 o azul, 0,0,1 o alguna otra combinación .

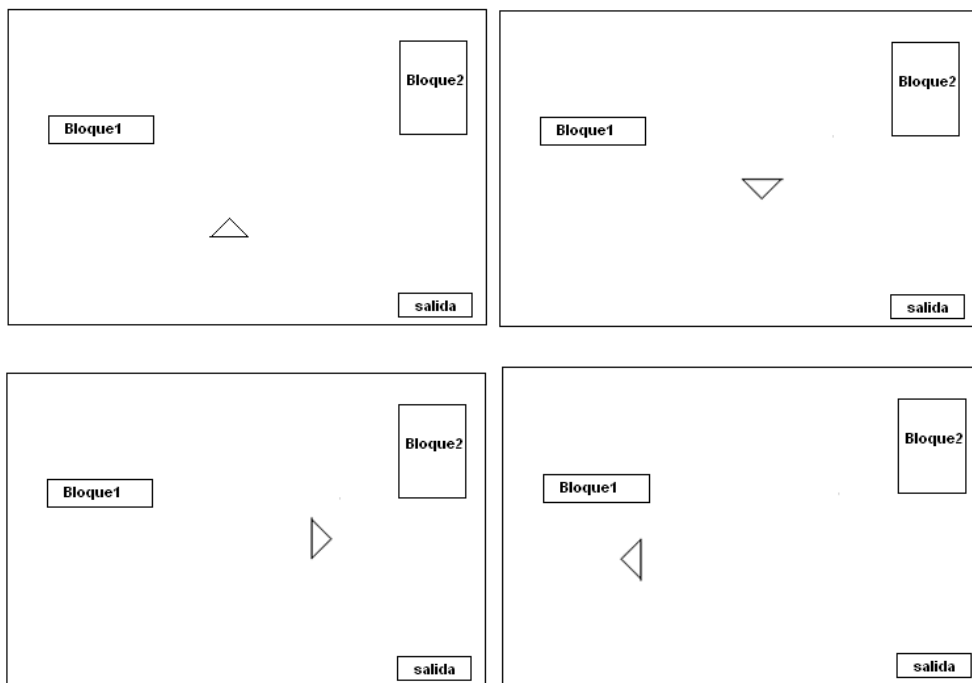
Con el fin de separar por color la salida de los bloques y la habitación.

Para cada dibujo que se escriba debe estar entre `glPushMatrix` y `glPopMatrix`, para que se muevan independientes.

*Comentario opcional sobre el push y pop: Si se dibuja un objeto 1 y luego un objeto 2 dentro del push y pop, y si desplazas el objeto 1, se desplaza el objeto 2, esto sirve para realizar por ejemplo articulaciones, dibujo un brazo, luego un antebrazo y luego una mano, si roto el antebrazo, que lo escribí primero, se mueve TODO el brazo.*

Para verificar si toco con algún obstáculo, se checa la posición x de mi espía contra la posición x de mi obstáculo, si son iguales, ya no puede avanzar, NUNCA debe ser mayor que el espacio de la habitación en x ni en y.

Ejemplos de movimientos del espía.



Las impresiones de pantalla y recogida de datos de usuario es con `printf` y `scanf`.

Para emular el movimiento de un ente es

1. Posición  $x_0, y_0$
2. Dibujar,
3. Mantener, esperar, dormir.
4. Borrar

Cambiar de posición  $x_1, y_1$  y repetir pasos a partir del 2. FIN DE EXPLICACION.

Entrega de proyecto por niveles.

**Nivel1: Utilizando procesos básicos.** Separa en procesos y programa por separado cada cosa.

Ejemplos:

programa1 que genere un contador con números aleatorios entre 30 y 60.

Programa2 que verifique que hay un obstáculo, si suponemos que obstáculo esta en x,y.

*Comentarios. Del nivel 1 se espera que tenga la mayoría de los programas que funcionen, por separado. El que mande más programitas que deben estar contenidos en el proyecto, estará en el mejor rango.*

**Nivel2a: Utilizando arreglos.** Localiza que puede hacerse con arreglos y envía los programas que funcionen por separado. Quien envíe más y que funcionen, estará en el mejor rango.

**Nivel2b: Utilizando gráficos opengl** . Quien realice los programitas de operación del nivel 1 y nivel2a ya funcionando dentro del programa en opengl, en el menor tiempo, estará en el mejor rango.

**Nivel3: avanzado.** Quien realice TODO el proyecto completo funcionando en el menor tiempo posible estará en el mejor rango.

Los rangos aparecerán en la pagina web correspondiente y/o por twitter.