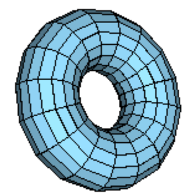
CLASE 16
EVENTOS Y CONDICIONALES EN SUPERFICIES PARAMÉTRICAS

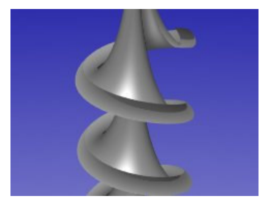
En la clase 11 practicamos con el uso de condicionales a través de algoritmos y la opción **dibujar-si**. En esta clase reforzaremos este concepto empleando el auxiliar llamado **evento**; igualmente, retomaremos el condicional **dibujar-si** con la combinación de conjunciones y disyunciones. Desarrollaremos una actividad muy interesante que aprovecha el potencial de Descartes en la representación de superficies paramétricas.

Actividad 1. Diseñar una escena múltiple que muestre varias superficies paramétricas, tales como las superficies curiosas de la botella de Klein y la cinta Moebius o el *donut* que llaman toro. Las ecuaciones que definen superficies como las anteriores se pueden consultar en algunas páginas de internet como:

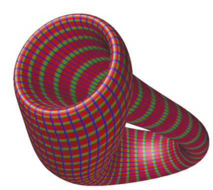
http://page.mi.fu-berlin.de/polthier/Events/vismath02/vgp/content/surfaces/. En esta página se presentan algunas superficies diseñadas con el software ***JavaView***: *"3D geometry viewer and a mathematical visualization software. The web-integration allows display of 3D geometries and interactive geometry experiments in any HTML document on the internet*".



http://www.econym.demon.co.uk/isotut/parametric.htm. Página diseñada por Mike Williams, que presenta un tutorial simple de cada superficie, así como las ecuaciones paramétricas que las defienen.



http://local.wasp.uwa.edu.au/~pbourke/geometry/. Página diseñada por Paul Bourke, que para una determinada superficie muestra una gran variedad de diseños.



Igualmente, hemos usado el software gratuito K3DSurf, que se puede descargas desde: http://k3dsurf.sourceforge.net/.

1.1 Crea un archivo con el nombre clase 16, añade una escena Descartes4 y elimina los botones de las esquinas de la escena. Para una representación más impactante usaremos un espacio 3D con fondo negro. Crea este espacio con una escala del 12% y despliegue pintor.

1.2 **Controles.** Para comprender mejor los controles que vamos a usar, empecemos diseñando la botella Klein. La curiosidad de la **botella de Klein** es que se trata de una superficie que no tiene ni interior ni exterior. Fue concebida por el matemático alemán Christian Felix Klein. Las ecuaciones paramétricas que permiten dibujarla son las siguientes:

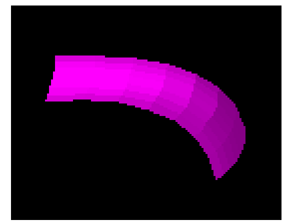
x = ((3*(1+sin(v)) + 2*(1-cos(v)/2)*cos(u))*cos(v))/3
y = ((4+2*(1-cos(v)/2)*cos(u))*sin(v))/3
z = (-2*(1-cos(v)/2) * sin(u))/3

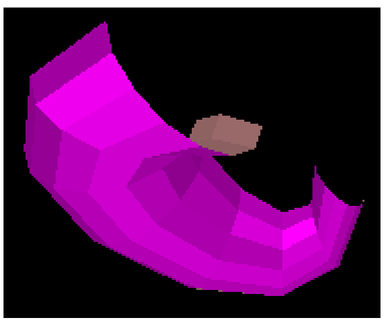
Para valores de **u** y **v** comprendidos entre -π y π.

Recordemos que se llaman paramétricas porque tanto **x**, como **y** y **z** dependen de los parámetros **u** y **v**. Para añadir una superficie, la ayuda del Nippe de Descartes dice:

> *Para superficies: debe tener la forma: x=X(u,v) y=Y(u,v) z=Z(u,v), donde X, Y y Z son expresiones numéricas dependientes de los parámetros u y v. La superficie consta de la red de cuadriláteros formada por los puntos: (X(i/Nu,j/Nv),Y(i/Nu,j/Nv),Z(i/Nu,j/Nv)) para i=0,...,Nu+1 y j=0,...,Nv+1. Antes de x, y, z se pueden definir variables intermedias que sólo se usan para los cálculos que se realizan al dibujar la superficie.*

Ahora crea una superficie cuyas ecuaciones sean las dadas para la botella de Klein. Al hacerlo te encontrarás con la superficie de la imagen derecha. Nada parecida a nuestra famosa botella. Esto era obvio, dado que en ninguna parte consideramos el rango de valores para **u** y **v,** que por definición es [-π, π].





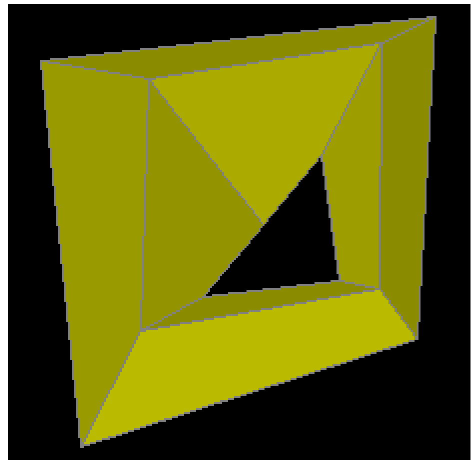
Cambiemos, entonces las ecuaciones por:

x=((3*(1+sin(pi*v)) + 2*(1-cos(pi*v)/2)* cos(pi*u))* cos(pi*v))/3
y=((4+2*(1-cos(pi*v)/2)*cos(pi*u))*sin(pi*v))/3
z=(-2*(1-cos(pi*v)/2) * sin(pi*u))/3

Observa que hemos incluido el factor π en cada parámetro **u** y **v.** Ahora la gráfica obtenida es más parecida a nuestra botella (ver imagen izquierda), pero aún le falta para completar la superficie ¿Qué le falta a las ecuaciones? Seguramente ya te habrás preguntado por la ausencia de -π. Sin embargo, si lees nuevamente, la ayuda del Nippe, notarás que los valores de **i** y de **j** son positivos. Es decir, no es posible iniciar desde -π ¿Cómo solucionar el problema? Hay dos formas, al menos para esta primera superficie.

La primera forma es utilizar el intervalo [0, 2π]. Cambia el factor en las ecuaciones anteriores de π por 2π, es decir, **pi** por **2*pi.** Con Nu = 30 y Nv = 30 y color amarillo; obtenemos, por fin, nuestra botella:

Si usáramos Nu = 4 y Nv = 4, podríamos visualizar 16 cuadriláteros que conforman la superficie. Recordemos la ayuda del Nippe: *"La superficie consta de la red de cuadriláteros formada por los puntos..."*. Igualmente, si activamos la casilla **aristas** se pueden apreciar dichos cuadriláteros con valores de Nu y Nv superiores a 4.



Sin embargo, el método usado para la botella Klein, cambiando el rango, no nos funcionará en todas las superficies con rangos negativos. Es allí, donde se sugiere un segundo método, el cual está enunciado en la ayuda del Nippe: "... *Antes de x, y, z se pueden definir variables intermedias que sólo se usan para los cálculos que se realizan al dibujar la superficie."*. En un principio no lo habíamos comprendido, hasta que nuestro Coordinador del Proyecto Descartes, José Román Galo Sánchez, nos lo aclaró.

> *"Si deseas que el parámetro **u** tome valores en [a, b] puedes introducir un cambio de variable U=a+(b-a)*u. Y si defines V=c+(d-c)*v tendremos que V tomará valores en [c, d]"*

Excelente la aclaración del Coordinador. Esto quiere decir que podemos usar las siguientes ecuaciones paramétricas:

U=a+(b-a)*u
V=c+(2*pi-c)*v
x=((3*(1+sin(V)) + 2*(1-cos(V)/2)*cos(U))*cos(V))/3
y=((4+2*(1-cos(V)/2)*cos(U))*sin(V))/3
z=(-2*(1-cos(V)/2) * sin(U))/3

Reemplaza las ecuaciones anteriores por estas nuevas y espera las siguientes instrucciones, antes de hacer clic en **aplicar**.

Observa con detenimiento los cambios de variable. En lugar de cuatro variables adicionales, usaremos tres (**a, b** y **c**). Podemos hacer variar **a** y **b** según nuestras necesidades. Para la botella Klein, entre $-\pi$ y π. Igualmente haremos con la variable **c**, la cual, en la mayoría de los casos, genera la superficie.

Así las cosas, continuemos con nuestra actividad. Crea los controles numéricos tipo barra **a**, **b** y **c**, con las siguientes características:

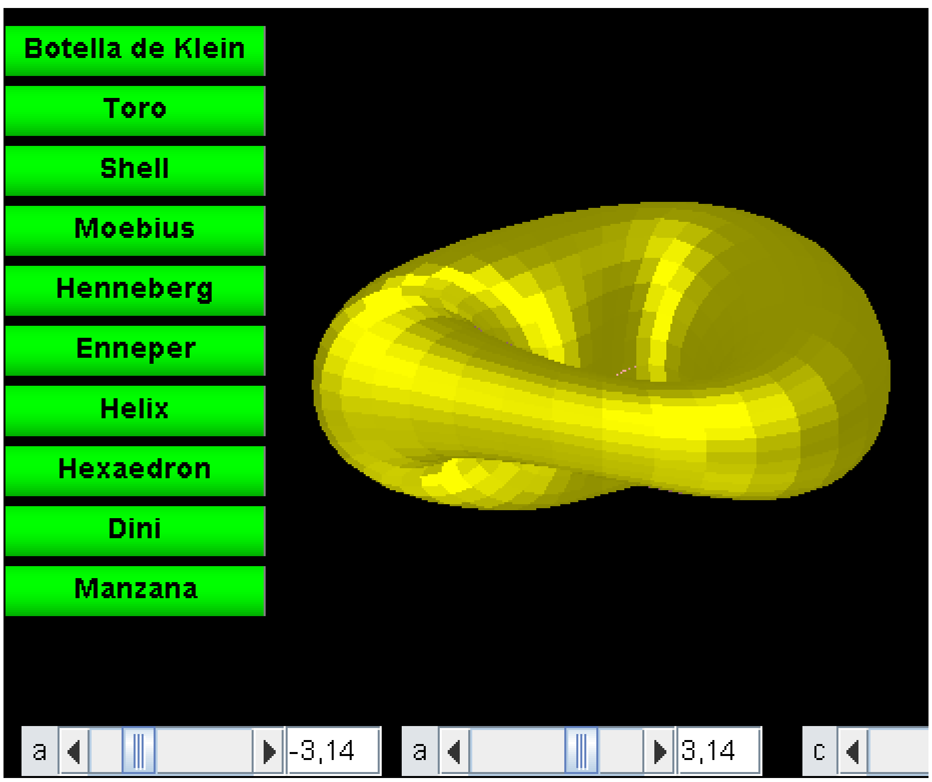
El control **a** con posición y tamaño, en el **interior de la escena**, definido por (10, 370, 180, 25) (ver imagen siguiente), valor de -pi, incremento de 0.1 y en el intervalo [-2pi, 2pi].

El control **b** con posición y tamaño en el **interior** definido por (200, 370, 180, 25), valor de pi, incremento de 0.1 y en el intervalo [-2pi, 5pi/2].

El control **c** con posición y tamaño en el **interior** definido por (400, 370, 180, 25), valor de 0, incremento de 0.1 y en el intervalo [-2pi, 2pi].

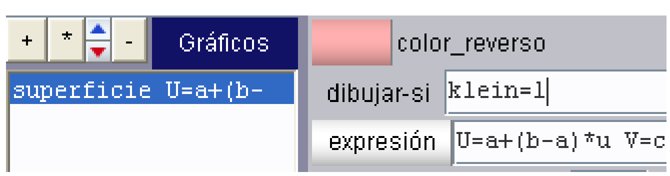
Los valores de los intervalos se calculan de tal forma que cubran las 10 superficies que pretendemos diseñar. Ahora haz clic en aplicar y presta atención a la siguiente advertencia: **Desplaza lentamente las barras** para que observes el efecto en la superficie; evita solapar superficies porque esto hará muy lenta la ejecución del Nippe. Continúa con los siguientes pasos, en los que se incluirán condicionales o eventos que eviten este solapamiento.

Ahora añadiremos 10 botones que permitan activar las superficies. Hazlo de tal forma que aparezcan como en la siguiente imagen. Por ejemplo, para el primer botón (botella de Klein) usamos posición y tamaño de (2, 20, 130, 25).

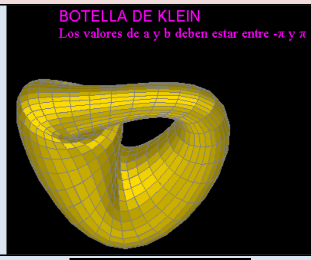
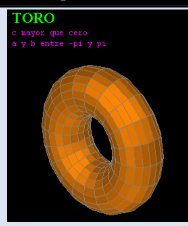
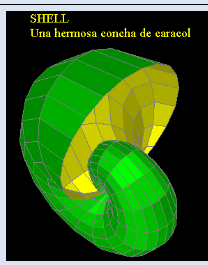
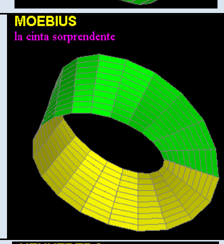
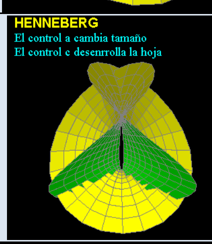
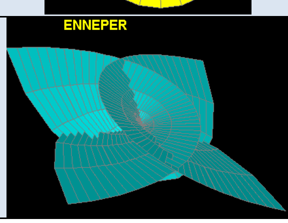


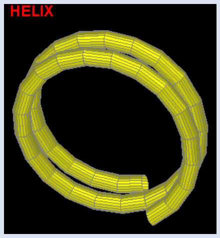
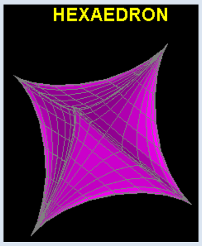
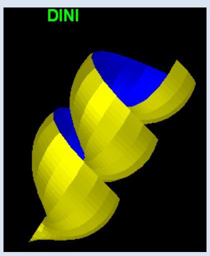
Queda como primer ejercicio completar los otros nueve botones. Luego regresaremos a ellos para su configuración.

En la tabla siguiente se muestran las diez superficies a representar con sus ecuaciones paramétricas. Se incluye, además, la condición para

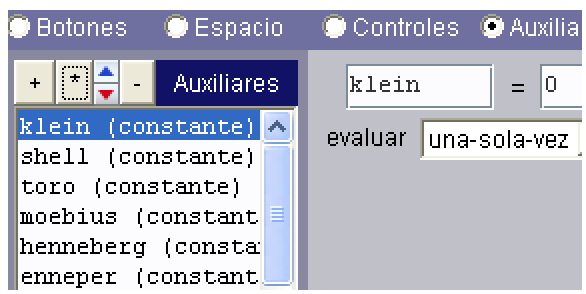


dibujarlas. Después de la tabla veremos el porqué de esta condición. Como ya ingresaste la primera superficie (la botella de Klein), sólo basta añadir la condición en **dibujar-si**; es decir, **Klein = 1**.

SUPERFICIE	ECUACIONES PARAMÉTRICAS	Dibujar-si
	U=a+(b-a)*u V=c+(2*pi-c)*v x=((3*(1+sin(V)) + 2*(1-cos(V)/2) * cos(U))* cos(V))/3 y=((4+2*(1-cos(V)/2)* cos(U)) *sin(V))/3 z=(-2*(1-cos(V)/2) * sin(U))/3	klein=1
	U=a+(b-a)*u V=c+(2*pi-c)*v x=(1+ 0.5*cos(U))*cos(V) y=0.5*sin(U) z=(1+ 0.5*cos(U))*sin(V)	toro=1
	U=a+(b-a)*u V=c+(2*pi-c)*v x=(1.2^V*(sin(U)^2 *sin(V))) y=(1.2^V*(sin(U)*cos(U))) z=(1.2^V*(sin(U)^2 *cos(V)))	(a>-0.1)&(b>-.78) &(shell=1)
	U=a+(b-a)*u V=c+(2*pi-c)*v x=1.5*(cos(V)+U*cos(V/2)*cos(V)) y=1.5*U*sin(V/2) z=1.5*(sin(V)+U*cos(V/2)*sin(V))	(moebius=1)& (abs(a+b)<.2)
	U=a+(b-a)*u V=c+(2*pi-c)*v x=(2*sinh(U)*cos(V) -(2/3)* sinh(3*U)*cos(3*V))/4 y=2*cosh(2*U)*cos(2*V)/4 z=(2*sinh(U)*sin(V) +(2/3)*sinh(3*U)*sin(3*V))/4	(henneberg=1)& (b>-0.07)&(b<0.08)
	U=a+(b-a)*u V=c+(2*pi-c)*v y=(U-U^3/3+U*V^2)/80 z=(U^2-V^2)/80 x=(V -V^3/3 + V*U^2)/80	enneper=1

	U=a+(b-a)*u V=c+(2*pi-c)*v x=1.8*((1-0.1*cos(V))*cos(U)) y=0.18*(sin(V) + U/1.7 -10) z=1.8*((1-0.1*cos(V))*sin(U))	(helix=1)&(c>-.01)
	U=a+(b-a)*u V=c+(2*pi-c)*v x=2*cos(V)^3*cos(U)^3 y=2*sin(U)^3 z=2*sin(V)^3*cos(U)^3	(hexaedron=1)& (c>3.13)&(b<6.3)& (a>-.3)&(a<-.2)
	U=a+(b-a)*u V=c+(2*pi-c)*v x=cos(2*U)*sin(V) y=((cos(V)+ln(tan(V/2)))+2*0.2*U) z=sin(2*U)*sin(V)	(dini=1)&(c>3)
	U=a+(b-a)*u V=c+(2*pi-c)*v x=cos(U) *(4 + 3.8* cos(V))/5 y=((cos(V) + sin(V) - 1)* (1 + sin(V)) *ln(1 - pi *V / 10) + 7.5 *sin(V))/5 z=sin(U) *(4 + 3.8* cos(V))/5	(manzana=1)& (b=0)&(c>-0.01)

1.3 **Auxiliares.** Comenzaremos utilizando 10 auxiliares constantes que corresponden a cada una de los nombres de las superficies, con valor inicial igual a cero (ver imagen derecha). Es decir: klein=0, shell=0, toro=0,…

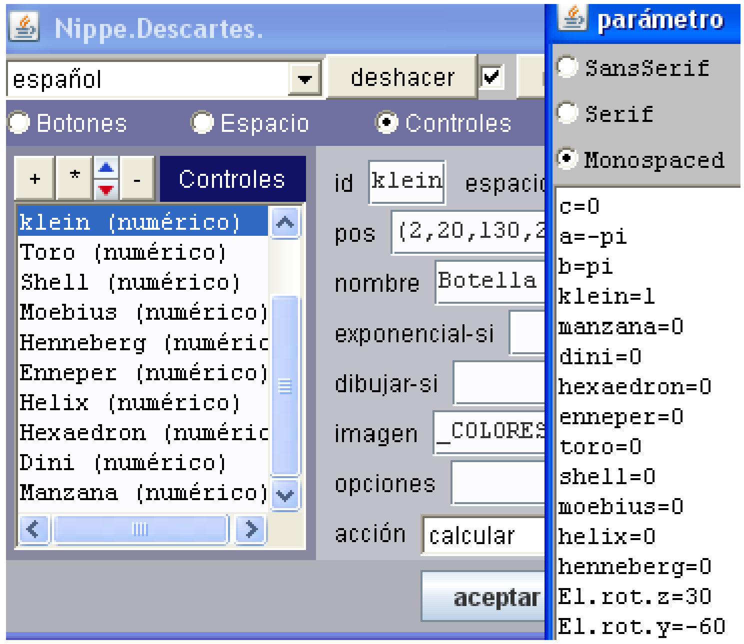


1.4 **Configuración de botones.** La idea es que se active una de las superficies al hacer clic sobre uno de los botones. Eso significa que podemos usar una de las constantes anteriores, cambiándolas al valor de uno cuando el botón correspondiente se activa y a cero cuando se activa otro (ver imagen siguiente). Igualmente, le asignamos los valores iníciales para cada superficie. Para el caso de nuestra botella Klein: a = -pi, b = pi y c = 0.

Veamos, entonces, como configuramos nuestro primer botón. El identificador del botón **(id)** será Klein, el nombre del botón será “Botella de Klein” y la acción será calcular.

Luego de ingresar esta configuración, haces clic en el botón **parámetro** para ingresar los datos que se ven en la imagen de la derecha.

Observa que los auxiliares para las demás superficies se dejan en cero, con el fin de desactivarlas, si fueron previamente activadas. De no hacerlo, puedes correr el riesgo que se superpongan las figuras. Igualmente se han dejado dos rotaciones iníciales, usa las que quieras (recuerda la clase de **rotini**).

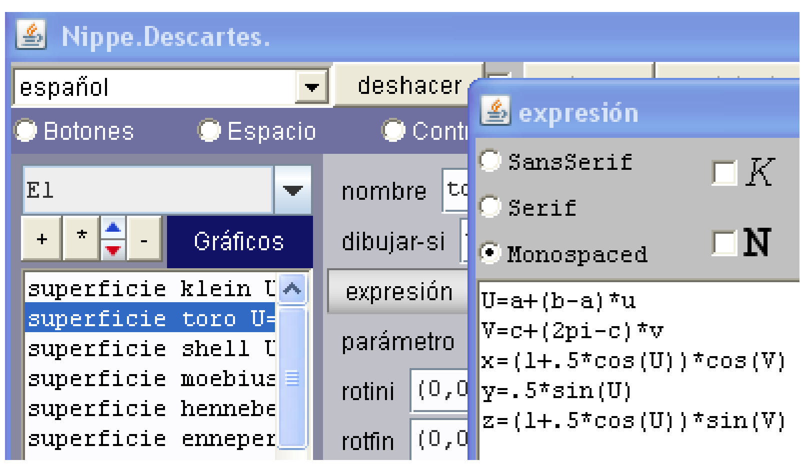


Sigue este procedimiento para las nueve superficies siguientes (segundo ejercicio), con estos valores iníciales:

Shell: c=-pi, a=0, b=pi
Moebius: c=0, a=-.37, b=.36
Henneberg: a=-.87, b=-0.06, c=0
Enneper: c=-2*pi, a=-2*pi, b=2*pi
Helix: c=0, a=-2*pi, b=2*pi
Toro: c=0, a=-pi, b=pi
Hexaedron: c=pi, a=-.25, b=2*pi
Dini: c=4.39, a=-3.01, b=4.6
Manzana: c=0, a=-2*pi, b=0

Obviamente, debes incluir la auxiliar correspondiente con un valor de uno y las demás con valor de cero.

1.5 **Superficies.** Ahora añadiremos las diez superficies (ver siguiente imagen). Las ecuaciones paramétricas son más fáciles de introducir haciendo clic en el botón **expresión**. En **dibujar-si** escribe las condiciones dadas en la tabla. Observa que estas



condiciones están vinculadas al auxiliar correspondiente. Activa aristas según tu gusto (ver imágenes de la tabla). Utiliza valores para Nu y Nv prudentemente, éstas hacen que se generen mayor o menor número de cuadriláteros con el consecuente uso de recursos del equipo, en general usa valores entre 10 y 30. Puedes usar *copy y paste* desde el archivo Word (Incluido en este tutorial).

Puedes incluir un texto descriptivo de la superficie, para ello debes usar el mismo condicional de la superficie (**dibujar-si**).

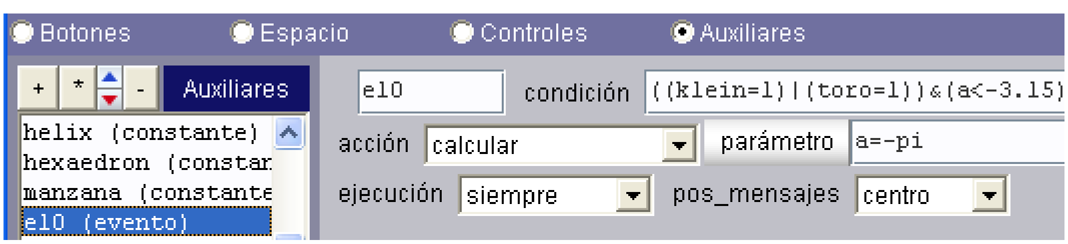
1.5 **Los Eventos.** Terminamos esta actividad con el concepto que nos ha convocado, el uso de eventos.

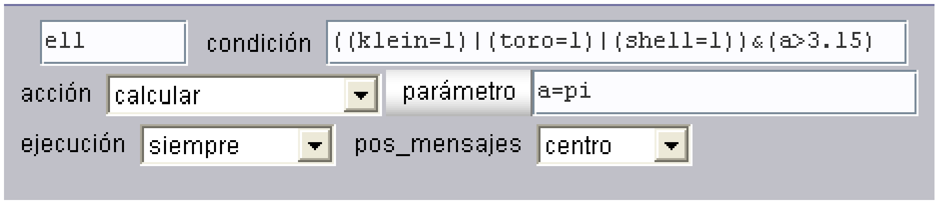
Si observas algunas de las condiciones de la tabla, se procura que la superficie no se muestre para valores por fuera de los rangos dados en el apartado 1.4. Lo anterior obedece a que podemos generar superficies superpuestas (de varias capas) que consumirán mucho recurso del equipo, ralentizando la escena y, a veces, bloqueando el *applet*.

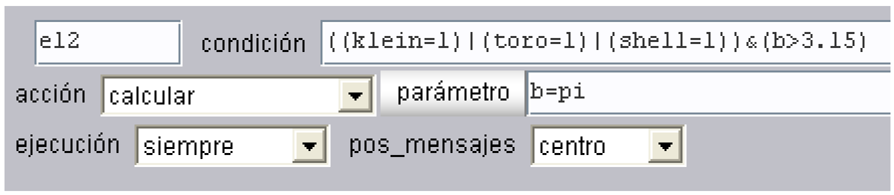
Esta opción de ocultar la imagen no es la más conveniente, didácticamente hablando. Lo mejor es usar una opción que impida el desplazamiento del control **a**, **b** o **c** por fuera del intervalo. He ahí una utilidad de los **eventos.**

Los eventos son auxiliares que de acuerdo a una condición, ejecutan una acción. Para nuestro caso, el evento sería aquella situación en la cual el usuario toma valores por fuera del rango; por ejemplo, para Klein o toro los valores de **a** menores que –pi. Si este evento se presenta, podemos programar una acción que regrese el valor de **a** a –pi. Veamos como:

Añade un auxiliar tipo evento, en condición escribirás: ((klein=1)|(toro=1))&(a<-3.15). El símbolo "**|**" representa el operador de disyunción y, el símbolo "**&**" el operador de conjunción. Es decir, la expresión anterior significa: "cuando se activa Klein **o** toro **y** a es menor que pi". En acción escogerás la opción **calcular** y en parámetro escribirás **a=-pi.** Esta acción garantizará que el valor de **a** nunca sea inferior a **pi.** En las imágenes siguientes observarás tres eventos. Trata de realizar otros para cambiar la condición de esconder imagen (tercer ejercicio).







Bueno, eso es todo por hoy, sólo te resta practicar lo enunciado.

Hasta la próxima