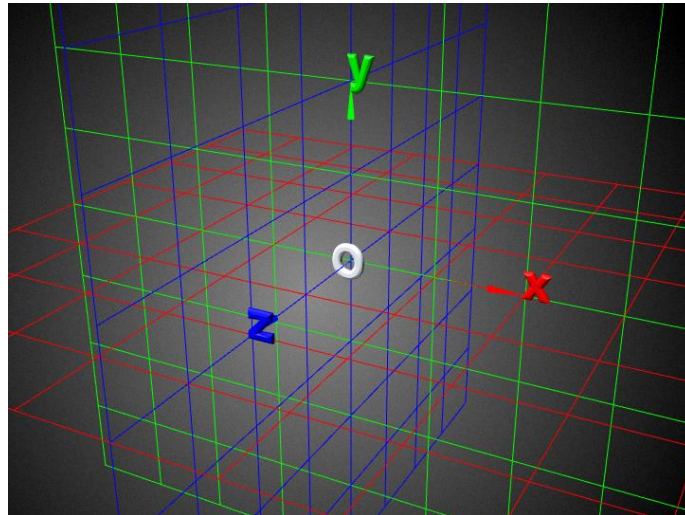


**Descargar la carpeta comprimida.** Ir a la dirección:

<http://conexionmatematica.catedu.es/virtual>

**Descomprimir la carpeta en una ubicación conocida.** El escritorio puede servir.

Al descomprimir el archivo virtual.zip se creará una carpeta llamada virtual. Entrando en la misma hay que abrir el fichero base.html con un navegador web, preferiblemente Firefox o Chrome. Si todo está bien se verá en el navegador una retícula con el origen O y los ejes X, Y y Z.



Los datos de este ejemplo están en el archivo data1.txt dentro de la carpeta ejemplos. Lo moveremos a la carpeta virtual y lo renombraremos como data.txt. Trabajaremos con un editor de texto sobre este archivo. En la carpeta se ha incluido Notepad2, un editor simple que sirve a nuestros propósitos. No se pueden utilizar editores como Word.

La estructura de data.txt es sencilla.

---

**data.txt**

```
function leerDatos() {  
  
    //variables globales  
    var num = -1, data = []; malla = true; opaco = 1;  
  
    //Inicio del bloque 1  
    ...  
    //Fin del bloque 1  
  
    //Inicio del bloque 2  
    ...  
    //Fin del bloque 2  
    ...  
  
    return data;  
  
}
```

---

El principio y el final, dentro de los recuadros, no se modifican. En la parte central se definen los bloques de piezas uno debajo de otro.

## Bloques.

Los bloques definen un conjunto de piezas que tienen el mismo perfil o contorno.

---

**data.txt**

//Inicio del bloque

num++; data[num] = {}; data[num].hole = []; data[num].mostrar = **true**;

//Escribe las coordenadas [x,y] del perfil vertical entre corchetes separadas por comas

```
var coor = [  
    [-1, 0], [-1, 1], [0, 1.5], [ 1, 1], [ 1, 0]  
];
```

//Escribe el color de la pieza en modo RGB y su grosor

```
var color = [  
    0xffffff  
];  
var grosor = [  
    0.5  
];
```

//Escribe las posiciones [x, y, z] donde mueves el perfil

```
var pos = [  
    [0, 0, 0]  
];
```

//Escribe los giros de la pieza [ejeX, ejeY, ejeZ]

```
var rot = [  
    [0, 0, 0]  
];
```

```
data[num].coor = coor; data[num].pos = pos; data[num].rot = rot; data[num].color =  
color; data[num].grosor = grosor;
```

//Fin del bloque

Las líneas que empiezan con // son comentarios que sirven de ayuda. Los valores más importantes que hay que cambiar son los números que representan coordenadas, posiciones o giros.

## Inicio y final de un bloque.

---

**data.txt**

//Inicio del bloque

num++; data[num] = {}; data[num].hole = []; data[num].mostrar = **true**;

Si no queremos ver las piezas definidas en este bloque, cambiaremos la palabra **true** por **false**, dejando el resto igual.

---

**data.txt**

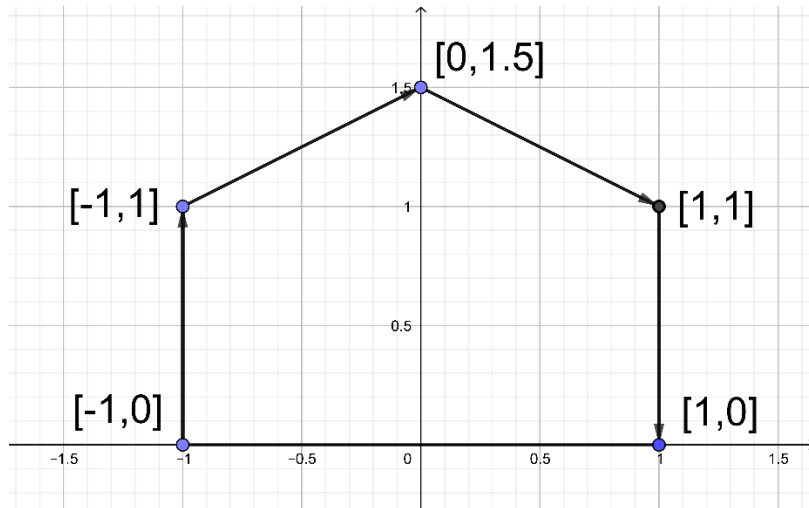
```
data[num].coor = coor; data[num].pos = pos; data[num].rot = rot;  
data[num].color = color; data[num].grosor = grosor;
```

//Fin del bloque

El final no hay que cambiarlo.

## Construcción del contorno.

Empezamos creando el perfil de la pieza. Utilizaremos el dibujo como ejemplo.



### data.txt

```
//Escribe las coordenadas [x,y] del perfil de la pieza entre corchetes separados por comas  
var coor = [  
    [-1, 0], [-1, 1], [0, 1.5], [ 1, 1], [ 1, 0]  
];
```

Las coordenadas de los puntos se colocan entre corchetes, no entre paréntesis como estamos acostumbrados. Dentro de cada corchete hay dos números separados por una coma, por ejemplo el primero es [-1, 0]

La notación de la coma decimal es el punto, no la coma. Si queremos escribir uno coma cinco pondremos 1.5

Se escriben las coordenadas de los puntos consecutivos que forman el contorno de la pieza en el sentido de las agujas del reloj. El resultado es una lista de corchetes separados por comas, salvo el último que no la lleva. [-1, 0], [-1, 1], [0, 1.5], [1, 1], [1, 0]

Por ultimo indicaremos el grosor y el color. Si queremos una pieza muy fina pondremos en grosor 0.01. En este caso el color es blanco y el grosor medio cuadro.

### data.txt

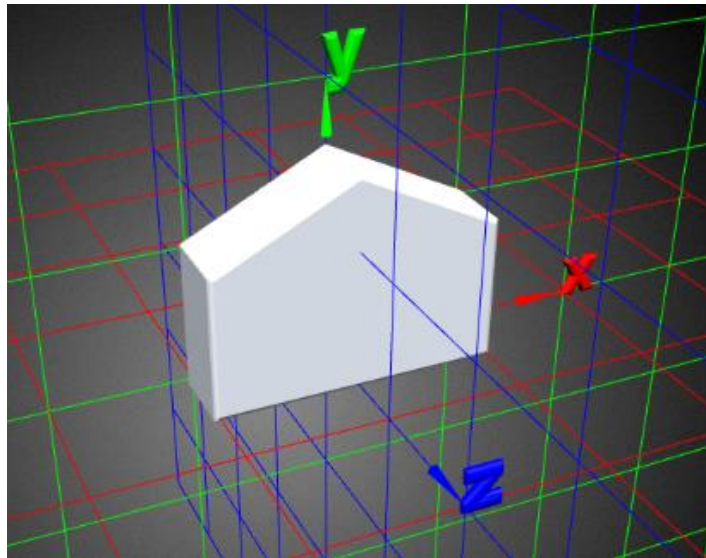
```
//Escribe los colores de cada pieza en modo RGB y su grosor  
var color = [  
    0xffffffff  
];  
var grosor = [  
    0.5  
];
```

Con los datos introducidos hasta ahora y estando  $[0, 0, 0]$  como la posición y los giros se concluye el bloque.

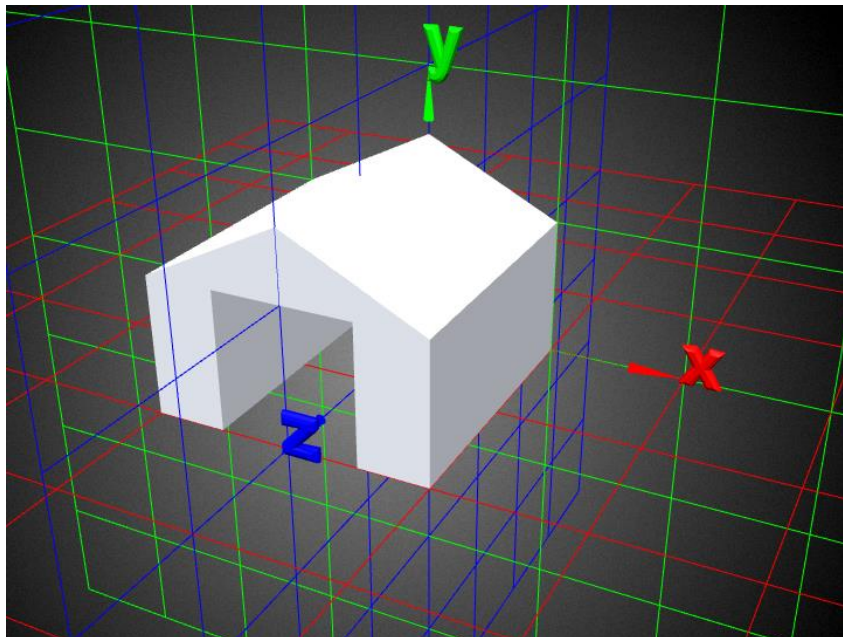
**data.txt**

```
//Escribe las posiciones [x, y, z] donde mueves el perfil  
var pos = [  
    [0, 0, 0]  
];  
//Escribe los giros de la pieza [ejeX, ejeY, ejeZ]  
var rot = [  
    [0, 0, 0]  
];
```

El resultado sería:



**Ejercicio.** Modifica el perfil para conseguir la pieza de la imagen inferior.



## Colores

Usamos la numeración hexadecimal para definir el color. Los dos primeros dígitos corresponden al canal rojo R, los dos siguientes al verde G y los dos últimos al azul B. Cada par oscila entre 00 y ff.

### Los tres colores básicos

Rojo	0xff0000	El canal de rojo está al máximo y los otros dos al mínimo
Verde	0x00ff00	El canal del verde está al máximo y los otros dos al mínimo
Azul	0x0000ff	El canal del azul está al máximo y los otros dos al mínimo
Blanco	0xffffffff	Los tres canales están al máximo

### Las combinaciones básicas

Amarillo	0xffff00	Los canales rojo y verde están al máximo
Cian	0x00ffff	Los canales verde y azul están al máximo
Magenta	0xff00ff	Los canales azul y rojo están al máximo
Gris claro	0xd0d0d0	Los tres canales tienen la misma intensidad
Gris oscuro	0x5e5e5e	Los tres canales tienen la misma intensidad