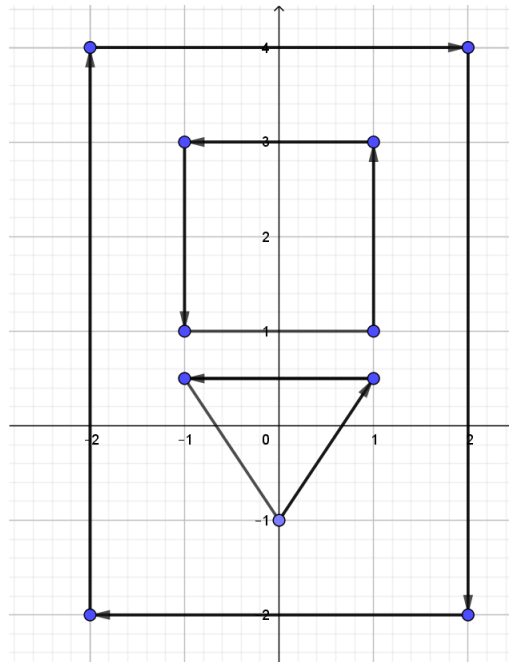


Huecos de un perfil.

Los datos de este ejemplo están en el archivo data4.txt dentro de la carpeta ejemplos. Lo moveremos a la carpeta virtual y lo renombraremos como data.txt. Usaremos el perfil se muestra en la imagen. El cuadrado y el triángulo son agujeros de la pieza rectangular.



Introducimos las coordenadas de los puntos del contorno exterior y no movemos a la pieza de su posición inicial ni la giramos.

data.txt

```
//Escribe las coordenadas [x,y] del perfil de la pieza entre corchetes separados por comas
```

```
var coor = [  
    [-2,-2],[-2,4],[2,4],[2,-2]  
];
```

```
//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]  
];
```

Mientras que los puntos del contorno exterior se recorren en sentido horario, los de los agujeros lo hay que recorrerlos en sentido contrario a las agujas del reloj.

Para crear los agujeros en una pieza hay que utilizar el bloque inferior, un poco menos amigable por mezclarse con código extraño.

data.txt

```
//Huecos
for (var k = 0; k <= 12; k++) data[num].hole[k] = [];
//Huecos de la pieza 1
var h = -1, hole = [];

var nPieza = [
  1
];
//Agujero 1
h++; hole[h] = [
  [1, 1], [1, 3], [-1, 3], [-1, 1]
];
//Agujero 2
h++; hole[h] = [
  [0, -1], [1, 0.5], [-1, 0.5]
];

for (var j = 0; j <= h; j++) {
  for (var k = 0; k < nPieza.length; k++) { var i = nPieza[k] - 1;
data[num].hole[i].push(hole[j]); }}
//Fin de los huecos de pieza 1
```

Indicamos el número de la pieza **1** en la que colocar los agujeros debajo de **var nPieza = [**. Escribimos las coordenadas del agujero con forma cuadrada en el sentido contrario a las agujas del reloj: [1, 1], [1, 3], [-1, 3], [-1, 1] debajo de la línea **h++; hole[h] = [**. Para el segundo agujero hacemos lo mismo. El código de cada hueco aparece en un recuadro.

data.txt

```
//Agujero 1
h++; hole[h] = [
  [1, 1], [1, 3], [-1, 3], [-1, 1]
];
```

```
//Agujero 2
h++; hole[h] = [
  [0, -1], [1, 0.5], [-1, 0.5]
];
```

El esquema de un bloque completo de piezas con algún agujero queda:

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 cada pieza [ejeX, ejeY, ejeZ]
var rot = [
    [0, 0, 0]
];

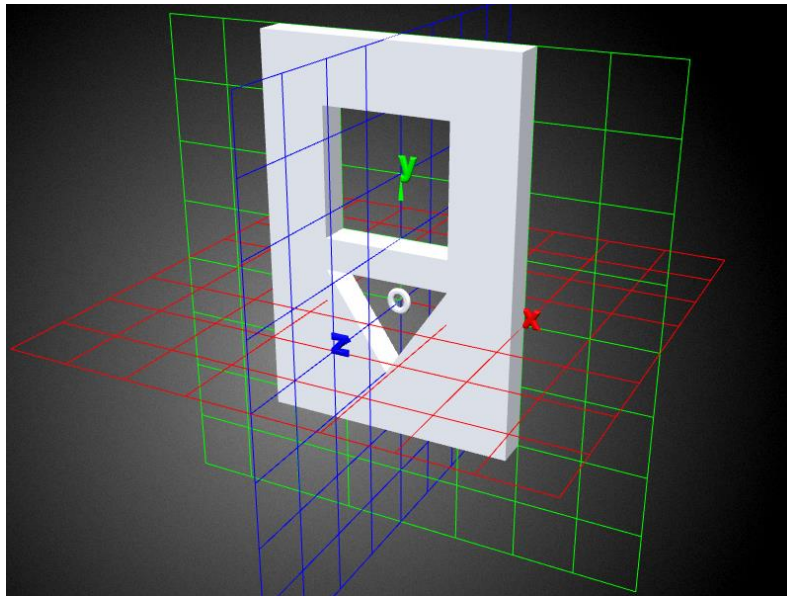
//Huecos
for (var k = 0; k <= 12; k++) data[num].hole[k] = [];
//Huecos de la pieza 1
var h = -1, hole = [];

var nPieza = [
    1
];
//Agujero 1
h++; hole[h] = [
    [1, 1], [1, 3], [-1, 3], [-1, 1]
];
//Agujero 2
h++; hole[h] = [
    [0, -1], [1, 0.5], [-1, 0.5]
];

for (var j = 0; j <= h; j++) {
    for (var k = 0; k < nPieza.length; k++) { var i = nPieza[k] - 1; data[num].hole[i].push(hole[j]); }
}
//Fin de los huecos de pieza 1

data[num].coor = coor; data[num].pos = pos; data[num].rot = rot; data[num].color = color;
data[num].grosor = grosor;
//Fin del bloque
```

El resultado final sería:



Ejercicio. Modifica el código para crear la pieza de la imagen inferior.

