

关于质心(Center of mass):

为了解决作业中的两个问题，先找到质心很重要，一种找到质心的方法是使用加权向量相加。

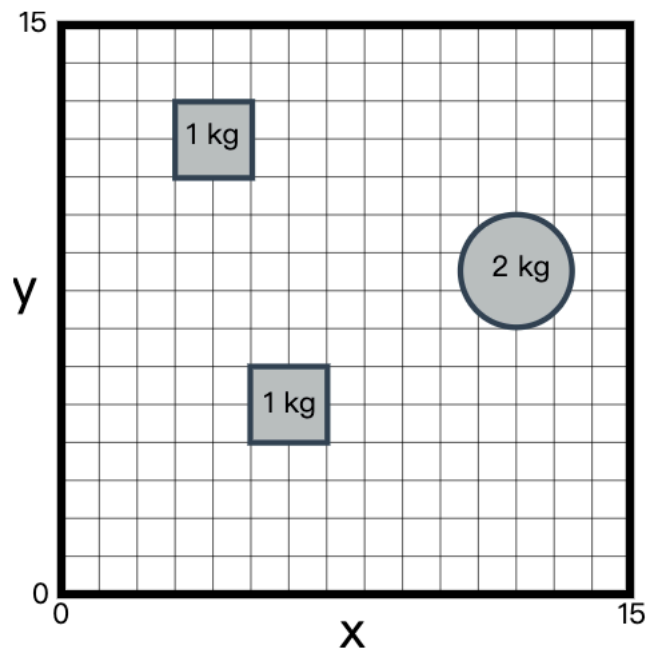
x轴:

$$\frac{m_1 \cdot x_1 + m_2 \cdot x_2 + m_3 \cdot x_3 + \dots}{m_1 + m_2 + m_3 + \dots}$$

Y轴:

$$\frac{m_1 \cdot y_1 + m_2 \cdot y_2 + m_3 \cdot y_3 + \dots}{m_1 + m_2 + m_3 + \dots}$$

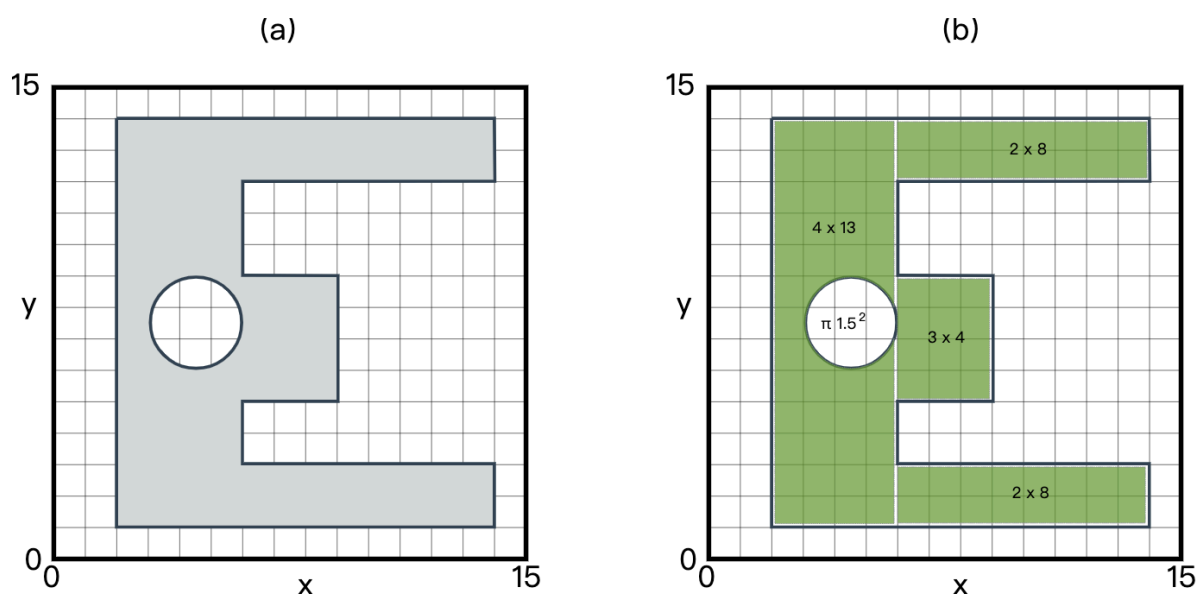
例如：找下面三个密度均匀的东西的质心：



则质心在x轴方向： $(1 \cdot 4 + 1 \cdot 6 + 2 \cdot 12) / (1 + 1 + 2) = 8.5$

质心在y轴方向： $(1 \cdot 5 + 1 \cdot 12 + 2 \cdot 8.5) / (1 + 1 + 2) = 8.5$

另一个例子：



上面这个不规则形状的物体，我们可以将这个对象分成四个矩形和一个圆，该材料具有均匀的密度，因此质量与面积成正比。为简单起见，我们可以以“平方”为单位表示每个部分的质量，如右图所示。

质心在x轴方向上：

$$\frac{16 \cdot 10 + 52 \cdot 4 + 12 \cdot 7.5 + 16 \cdot 10 + (-7.1) \cdot 4.5}{16 + 52 + 12 + 16 - 7.1} = 6.6$$

质心在y轴方向上：

$$\frac{16 \cdot 13 + 52 \cdot 7.5 + 12 \cdot 7.0 + 16 \cdot 2 + (-7.1) \cdot 7.5}{16 + 52 + 12 + 16 - 7.1} = 7.4$$

更详细的可见：<https://www.khanacademy.org/science/physics/linear-momentum/center-of-mass/a/what-is-center-of-mass>

关于理解数组

下面这个array表示了一个立方体，其中13这个小立方体是从哪个面都看不到的，因为在最中心，array可以理解成一个大大的立方体，分成了27个小立方体的组合，array里的数字，就是每个小立方体的值。13是最最中心的那个立方体。

```
|: 1 # a cube example
    2 c = np.arange(27).reshape(3, 3, 3)
    3 c
```

```
|: array([[[ 0,  1,  2],
           [ 3,  4,  5],
           [ 6,  7,  8]],

          [[ 9, 10, 11],
           [12, 13, 14],
           [15, 16, 17]],

          [[18, 19, 20],
           [21, 22, 23],
           [24, 25, 26]]])
```

但是下面这个其实也是三维数组

```
1 turned = np.rot90(a, k=1, axes=(1, 0))
2 turned.shape
```

(3, 4, 2)

```
1 turned
```

```
array([[[18, 19],
        [12, 13],
        [ 6,  7],
        [ 0,  1]],

       [[20, 21],
        [14, 15],
        [ 8,  9],
        [ 2,  3]],

       [[22, 23],
        [16, 17],
        [10, 11],
        [ 4,  5]]])
```

怎么理解呢？

首先，对下面这个黑色图中A形状来说, A【1, 1】=42,这个不难理解。

但是对于右图右边B形状这种立体来说，其实是可以类似用类似于左边白色图这种的二维数组来表示这个立方体。下面左边白图代码的绿色部分是表示一个立方体的第二层，整个数组代表了一个立方体的四层情况（那个立方体是个2*3*4的立方体）。

对于B立方体来说，0 ,na ,8,1就是B立方体的第一层（就是最顶上那一层）

```
file Edit View Insert Cell Kernel Widgets Help
+ < > Run Code
Out[96]: 18

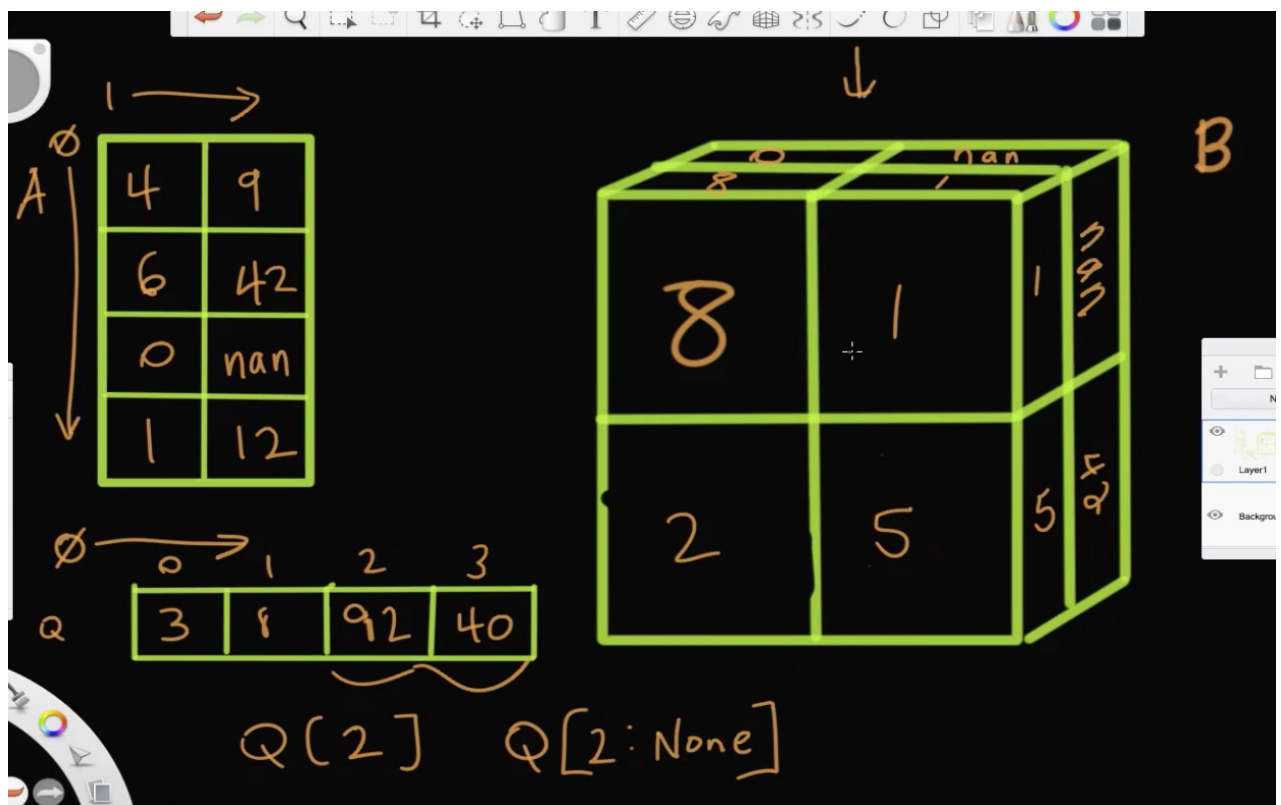
In [97]: 1 a
Out[97]: array([[ 0,  1],
               [ 2,  3],
               [ 4,  5]],

              [[ 6,  7],
               [ 8,  9],
               [10, 11]],

              [[12, 13],
               [14, 15],
               [16, 17]],

              [[18, 19],
               [20, 21],
               [22, 23]])

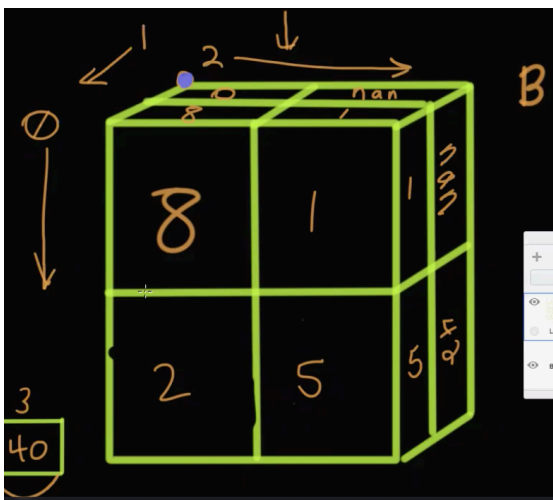
In [98]: 1 a[:,1,:]
```



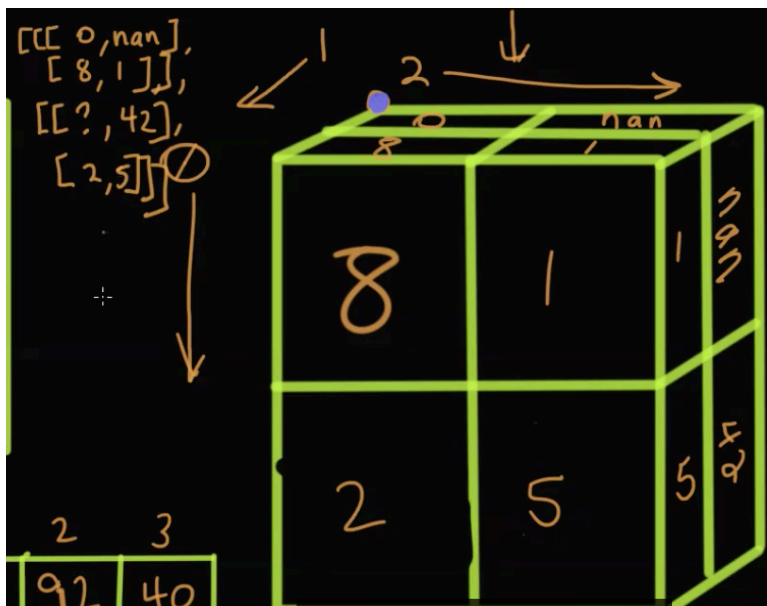
下面用B立方体来解释：

如图所示，竖着的垂直的轴是0来表示，就是从顶部到底部，1轴是从里到外（垂直于电脑屏幕），2轴就是向右边延伸的。紫色的小点点是原点（记住这是原点）。

举个例子：对于侧面的那个42的小立方体来说，它的坐标就是【1，0，1】，因为与紫色的那个原点比，0轴有延伸，所以是1；1轴没有延伸，所以是0；2轴有延伸，所以是1，合起来就是1，0，1



这个B正方体整体也可以用左上角黄色笔记这么表示。“？”是代表了看不到那个小立方体：



下面是rotation,旋转

使用numpy来实现，下面这个代码里，a是要旋转的数组，axes是指的旋转的方向，k=1指的是有多少个单元，k=1指的是有90度，k=2有180度，k=3有270度，k=3和k=-1的意义一样。

理解axes= (2, 1) ，需要又回到之前的那个B立方体，紫色的那个原点作为旋转点，(2, 1) 的含义是，之前说过2轴和1轴，(2, 1) 就是指明了旋转所垂直的面在2轴和1轴组成的面上，也就是说，这个立方体会绕着与0轴平行的一条线旋转。(2, 1) 和 (1, 2) 的意义是一样的。

```
In [101]: 1 r = np.rot90(a, axes=(2, 1), k=1)
           2 r
Out[101]: array([[ 4,  2,  0],
                  [ 5,  3,  1]],
                [[10,  8,  6],
                  [11,  9,  7]],
                [[16, 14, 12],
                  [17, 15, 13]])
```

所以：

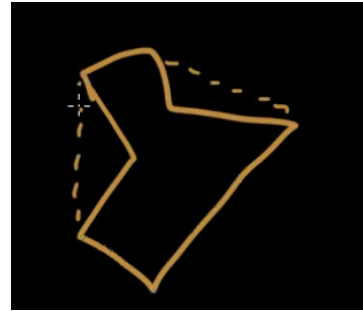
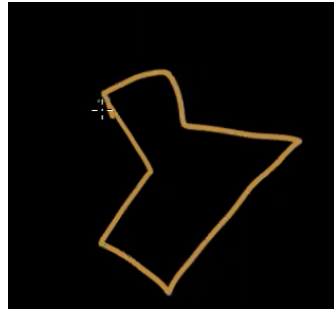
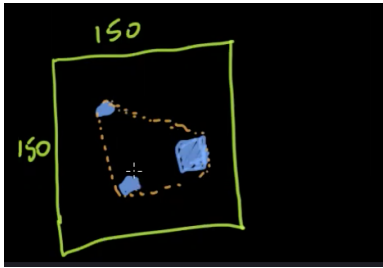
下图左边这个数组，在经过旋转后，可以变成右边这个数组。很多数字的位置变了

<pre>array([[0, 1], [2, 3], [4, 5]], [[6, 7], [8, 9], [10, 11]], [[12, 13], [14, 15], [16, 17]], [[18, 19], [20, 21], [22, 23]])</pre>	<pre>array([[4, 2, 0], [5, 3, 1]], [[10, 8, 6], [11, 9, 7]], [[16, 14, 12], [17, 15, 13]], [[22, 20, 18], [23, 21, 19]])</pre>
---	---

中心就是重心，就像人做不同的动作，人的重心就不一样，比如人在站立的时候，人的重心在地上的投影是两脚之间，人在弯腰的时候，人的中心在地上的投影是腹部与地面的垂线处。

一个重量分布均匀的石块，且在地上的投影是正方形的话，那么他的中心就是正方形中心。但是重量分布不均匀的石块就不是这样了。在这次作业中，前面说的质心就是重心。

什么是convex hull,想象一下有个正方形模板，我们在上面打上蓝色的钉子，如下图，橙色区域的形状连接起来的部分就是这个convex hull。如果convex hull是在二维平面上的话，不管钉子有多少个，convex hull都是个多边形，而不会像中间这个图中橙色线条画出来的一样呈现一个奇奇怪怪的形状。对中间错误convex hull的修正，是像右边这个图虚线一样的画法才是正确的。



如果质心投影在下图的A点这个东西就可以被放得很稳，如果在B点就会不稳（物理知识）。所以确定一个东西放得稳不稳，可以先计算物体底部的convex hull的多边形的形状，然后检查物体的质心是在convex hull里面还是外面。

