

In [2]:

```
##### HW1-1
from PIL import Image
imgname1 = r'C:\Users\apple\OneDrive\桌面\影像\HW1\laptop_left.png'
img1 = Image.open(imgname1)
imgname2 = r'C:\Users\apple\OneDrive\桌面\影像\HW1\laptop_right.png'
img2 = Image.open(imgname2)
w1 , h1 = img1.size  ##200/356
w2 , h2 = img2.size  ##280/356

new_w = w1+w2
new_h = h1
new_img = Image.new( 'RGBA' , (new_w,new_h))

new_img.paste(img1,(0,0))
new_img.paste(img2,(w1,0))

new_img.save( 'new.png' )
```

In [13]:

```
##### HW1-2
from PIL import Image
import math

im_name = r'C:\Users\apple\OneDrive\桌面\影像\HW1\new.png'
im = Image.open(im_name)

# 計算旋轉後圖片的大小
theta = math.radians(15)
w, h = im.size
new_w = round(abs(w * math.cos(theta)) + abs(h * math.sin(theta)))
new_h = round(abs(h * math.cos(theta)) + abs(w * math.sin(theta)))

new_im = Image.new("RGB", (new_w, new_h), color=(255, 255, 255))

# 計算圖片的中心點
cx, cy = w // 2, h // 2

# 計算旋轉矩陣
cos_theta = math.cos(theta)
sin_theta = math.sin(theta)
rotation_matrix = ((cos_theta, sin_theta), (-sin_theta, cos_theta))

pixels = im.load()
new_pixels = new_im.load()

for i in range(new_w):
    for j in range(new_h):
        # 計算原始圖片上對應的坐標
        x, y = i - new_w // 2, j - new_h // 2
        new_x = round(x * cos_theta + y * sin_theta) + cx
        new_y = round(-x * sin_theta + y * cos_theta) + cy

        # 檢查像素是否超出圖片範圍
        if new_x >= 0 and new_x < w and new_y >= 0 and new_y < h:
            # 設置新圖片的像素值
            new_pixels[i, j] = pixels[new_x, new_y]

new_im.save('rotate.png')
```

In [11]:

```
##### HW1-3
lena_name = r'C:\Users\apple\OneDrive\桌面\影像\HW1\lena.bmp'
lena = Image.open(lena_name)

w , h = lena.size #[512,512]

new_w, new_h = 1024, 1024

hw3 = Image.new("RGB", (new_w, new_h), color=(255, 255, 255))

#縮放比例
sx = new_w / w
sy = new_h / h

pixels = lena.load()
new_pixels = hw3.load()

for i in range(new_w):
    for j in range(new_h):
        # 計算對應的原始圖像的坐標
        x = i / sx
        y = j / sy

        # 計算四個最近的像素的坐標
        x1, y1 = int(x), int(y)
        x2, y2 = x1 + 1, y1 + 1

        # 計算插值權重
        wx = x - x1
        wy = y - y1

        try:
            r1, g1, b1 = pixels[x1, y1]
            r2, g2, b2 = pixels[x2, y1]
            r3, g3, b3 = pixels[x1, y2]
            r4, g4, b4 = pixels[x2, y2]
        except IndexError:
            # 如果超出圖像範圍 · 則將像素設置為白色
            r1, g1, b1 = r2, g2, b2 = r3, g3, b3 = r4, g4, b4 = 255, 255, 255

        # 計算插值後的像素值
        r = int((1 - wx) * (1 - wy) * r1 + wx * (1 - wy) * r2 + (1 - wx) * wy * r3 + wx
        g = int((1 - wx) * (1 - wy) * g1 + wx * (1 - wy) * g2 + (1 - wx) * wy * g3 + wx
        b = int((1 - wx) * (1 - wy) * b1 + wx * (1 - wy) * b2 + (1 - wx) * wy * b3 + wx

        # 設置新圖像的像素值
        new_pixels[i, j] = (r, g, b)

hw3.save('lena2.bmp')
```

In [4]:

```
##### HW1-4
lena_name = r'C:\Users\apple\OneDrive\桌面\影像\HW1\lena.bmp'
lena = Image.open(lena_name)

graveler_name = r'C:\Users\apple\OneDrive\桌面\影像\HW1\graveler.bmp'
graveler = Image.open(graveler_name)
gra = graveler.convert('RGBA')

pixels = gra.load()

for i in range(gra.size[0]):
    for j in range(gra.size[1]):
        r,g,b,a = pixels[i,j]
        if (r>=240 and g >= 240 and b >= 240 and a >= 240):
            pixels[i,j]=(255,255,255,0)

lena.paste(gra,(200,200),mask=gra)
lena.save('lena3.bmp')
```

Out[4]:

(104, 53)