Part 1: Gaussian Filtering
1.boxfilter(n)
(1) n = 3

```
[[0.33333333 0.33333333 0.33333333]
 [0.33333333 0.33333333 0.33333333]
 [0.33333333 0.33333333 0.33333333]]
```

(2) n= 4

```
 Traceback (most recent call last):
   File "A1final.py", line 18, in <module>
     print(boxfilter(n))
   File "A1final.py", line 10, in boxfilter
     assert mod != 0, "Dimension must be odd."
 AssertionError: Dimension must be odd.
```

(3) n = 5

```
[[0.2 0.2 0.2 0.2 0.2]
 [0.2 0.2 0.2 0.2 0.2]
 [0.2 0.2 0.2 0.2 0.2]
 [0.2 0.2 0.2 0.2 0.2]
 [0.2 0.2 0.2 0.2 0.2]]
```

2.gauss1d(sigma)
(1)sigma = 0.3

```
    [0.00141816 0.99716368 0.00141816]
```

(2)sigma = 0.5

```
 [4.99999916e-01 7.61498859e-09 1.52951135e-07 7.61498859e-09
  4.99999916e-01]
```

(3)sigma = 1

```
 [0.12388227 0.1321298  0.07528529 0.33740527 0.07528529 0.1321298
  0.12388227]
```

(4)sigma = 2

```
[0.06904283 0.06904283 0.06904283 0.06904283 0.06905968 0.06093008
 0.18767786 0.06093008 0.06905968 0.06904283 0.06904283 0.06904283
 0.06904283]
```

3.sigam = 3

5.Since the expression can be
write to the form of two identical function for x and for y multiply together.

$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2}\exp^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$= \underbrace{\left(\frac{1}{\sqrt{2\pi}\sigma}\exp^{-\frac{x^2}{2\sigma^2}}\right)}_{\text{function of x}}\underbrace{\left(\frac{1}{\sqrt{2\pi}\sigma}\exp^{-\frac{y^2}{2\sigma^2}}\right)}_{\text{function of y}}$$

We can simply just implement a gauss1d(sigma) function and for each pixel(x,y), apply it to x and y separately, then multiply the result together to get the 2d gaussian filter. By doing this our runtime original from m^2*n^2, improve to 2*m*n^2(where m stand for the height of filter, n stand for the height of pixel)

Part 2:
1.sigam = 3

2.sigam = 3

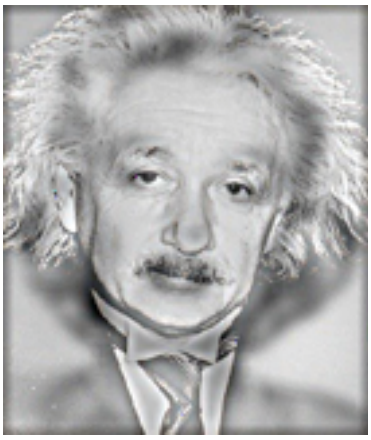
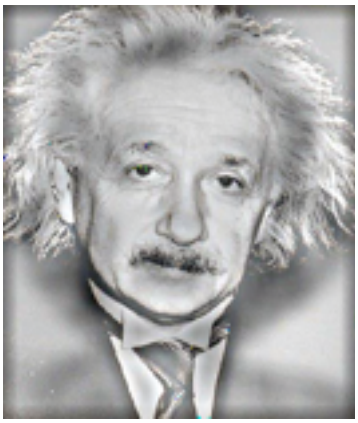
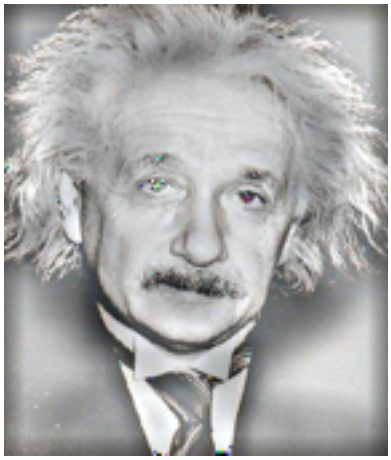
3.sigam = 3

sigam = 2

sigam = 3

sigam = 4

sigam = 2,3,4

sigam = 2,3,4



sigam = 4, 5,6