Q.1 $F = \text{convolve2d}(g, h, \text{mode} = \text{'valid'})$

$$F = \begin{bmatrix} 0 & -52 \\ 45 & -2 \end{bmatrix}$$

P2. $\sigma = 1$

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

$$MSE = \frac{1}{n} \sum_{i,j} \left(G(i,j) - \hat{G}(i,j) \right)^2$$

$\Rightarrow 7.08 \times 10^{-6}$

P3. $\nabla^2 F = \dfrac{\partial^2 F}{\partial x^2} + \dfrac{\partial^2 F}{\partial y^2}$

Kernel $= \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$

1. $\dfrac{\partial^2 F}{\partial x^2} \approx F(i+1, j) - 2F(i, j) + F(i-1, j)$

$\dfrac{\partial^2 F}{\partial y^2} \approx F(i, j+1) - 2F(i, j) + F(i, j-1)$

From a kernel point of view

$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$

Center Pixel weight $= -4$
Neighboring pixels x and y weight $= 1$

P. 4. $G(x,y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$

a) $F(x,y) = F(x)\, F(y)$ for a separable filter

2D Gaussian.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2}{2\sigma^2}} e^{-\frac{y^2}{2\sigma^2}}$$

$$G(x,y) = \left(\frac{1}{\sqrt{2\pi}\,\sigma} e^{-x^2/2\sigma^2}\right)\left(\frac{1}{\sqrt{2\pi}\,\sigma} e^{-y^2/2\sigma^2}\right)$$

$$G(x,y) = G_x(x) \cdot G_y(y)$$

2D Gaussian is separable

6) $\nabla^2 G(x,y) = \left(\frac{x^2+y^2}{\sigma^4} - \frac{2}{\sigma^2} \right) G(x,y)$

$$\frac{\partial G}{\partial x} = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{(x^2+y^2)}{2\sigma^2}} \cdot \left(\frac{-x}{\sigma^2} \right)$$

$$\frac{\partial G}{\partial x} = \frac{-x}{\sigma^2} \; G(x,y)$$

$$\frac{\partial^2 G}{\partial x^2} = \frac{-1}{\sigma^2} G(x,y) + \frac{x^2}{\sigma^4} G(x,y)$$

$$\frac{\partial^2 G}{\partial x^2} = \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right) G(x,y)$$

y will be similar

$$\frac{\partial^2 G}{\partial y^2} = \left(\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right) G(x,y)$$

$$\nabla^2 G(x,y) = \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right) G(x,y) + \left(\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2}\right) G(x,y)$$

$$\nabla^2 G(x,y) = \left(\frac{x^2+y^2}{\sigma^4} - \frac{2}{\sigma^2}\right) G(x,y)$$

```
[1] import numpy as np
    import math

    # Template approximation (scaled)
    template = np.array([
        [1, 4, 7, 4, 1],
        [4, 16, 26, 16, 4],
        [7, 26, 41, 26, 7],
        [4, 16, 26, 16, 4],
        [1, 4, 7, 4, 1]
    ]) / 273

    # Define the Gaussian function
    def gaussian_2d(x, y, sigma=1):
        return (1 / (2 * np.pi * sigma**2)) * np.exp(-(x**2 + y**2) / (2 * sigma**2))

    # true Gaussian values -2 to 2
    x_vals = np.arange(-2, 3)
    y_vals = np.arange(-2, 3)
    true_gaussian = np.array([[gaussian_2d(x, y) for y in y_vals] for x in x_vals])

    mse = np.mean((true_gaussian - template) ** 2)
    mse
```

```
7.07642477467327e-06
```