SOLUTIONS WEEK 1

Solution 1.1. The answer here is the Gaussian distribution due to the Central Limit Theorem. Note that $\mathbb{E}[U_i] = 1/2$ and $\mathrm{Var}(U_i) = 1/12$. Therefore, we can define the sample mean

$$\bar{X} = \frac{1}{12}\sum_{i=1}^{12} U_i,$$

and by the CLT, we have

$$\frac{(\bar{X} - \mu)}{\sigma/\sqrt{n}} \xrightarrow{d} \mathcal{N}(0, 1).$$

Writing this out, we can see that the lhs is given by

$$\frac{(\bar{X} - \mu)}{\sigma/\sqrt{n}} = \frac{\frac{1}{12}\sum_{i=1}^{12} U_i - \frac{1}{2}}{\frac{1}{\sqrt{12}}/\sqrt{12}} = \sum_{i=1}^{12} U_i - 6.$$

This shows that this random variable $X = \sum_{i=1}^{12} U_i - 6$ is approximately Gaussian. We can verify this by sampling U_i's and computing X and plotting the histogram. The code is given below:

Solution 1.2. The density is Gaussian. For both the above solution and drawing samples from this density and plotting against `scipy.stats.norm`, see the code below.

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats

# Explicit example
# Let us do for explicitness this in a for loop.

rng = np.random.default_rng(5)

N = 10000
X = np.zeros(N)

for i in range(N):
    U_vec = rng.uniform(0, 1, 12)
    X[i] = np.sum(U_vec) - 6

plt.hist(X, bins=100, density=True)
xx = np.linspace(-5, 5, 1000)
plt.plot(xx, stats.norm.pdf(xx, 0, 1)) # Plot the normal distribution (or see lecture)
plt.show()
```

Solution 1.3. The CDF of X is given by

$$\begin{aligned}
F_X(x) = \mathbb{P}(X \leq x) &= \mathbb{P}(-\lambda^{-1}\log U \leq x), \\
&= \mathbb{P}(\log U \geq -\lambda x), \\
&= \mathbb{P}(U \geq \exp(-\lambda x)), \\
&= 1 - \mathbb{P}(U \leq \exp(-\lambda x)), \\
&= 1 - \exp(-\lambda x).
\end{aligned}$$

Taking the derivative of this, one can see that

$$p_X(x) = \lambda \exp(-\lambda x),$$

which is the exponential density. The code is given below.

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats

rng = np.random.default_rng(12345)

N = 100000
X = np.zeros(N)
lam = 1

for i in range(N):
    U = rng.uniform(0, 1)
    X[i] = -lam * np.log(U)

xx = np.linspace(0, 10, 1000)
plt.hist(X, bins=100, density=True, label="Histogram")
plt.plot(xx, lam * np.exp(-lam * xx), label="Exponential PDF")
plt.xlim(0, 10)
plt.legend()
plt.show()
```