AAG03

- Desenvolver no Jupyter Notebook exemplos (enunciados e resoluções) que expliquem cada uma das distribuições apresentadas na aula de hoje:
    - Bernoulli
    - Binomial
    - Poisson
    - Geométrica
    - Uniforme
    - Exponencial
    - Gaussiana (Normal)
- Regras:
    1. Código e resultados devem ser explicados em Markdown com comandos LaTeX.
    2. Os formatos de entrega devem ser `.pdf` e `.ipynb` (código fonte + markdowns).

Distribuições de variáveis aleatórias discretas:

- Bernoulli
- Binomial
- Gométrica
- Poisson

Distribuição Bernoulli

A distribuição Bernoulli é usada para modelar experimentos que possuem dois possíveis resultados: "sucesso" (com probabilidade (p)) e "fracasso" (com probabilidade $(1-p)$).

A função de probabilidade é definida como:

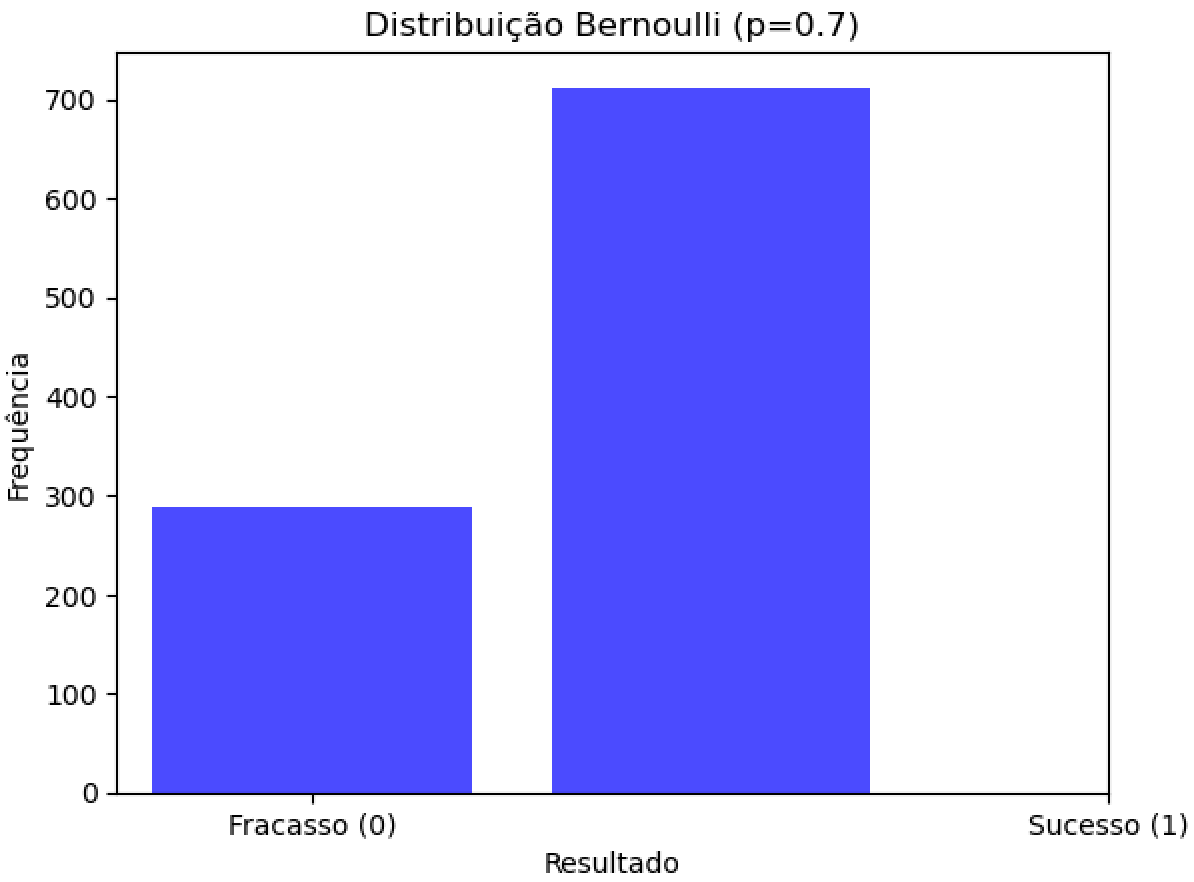
$$P(X=x)=\begin{cases} p & \text{se } x=1, \\ 1-p & \text{se } x=0 \end{cases}$$

Onde:

- $0 \leq p \leq 1$: probabilidade de sucesso.

In [14]:
```
import numpy as np
import matplotlib.pyplot as plt

# Parâmetros
p = 0.7  # Probabilidade de sucesso
n = 1000  # Número de simulações

# Simulação
data = np.random.binomial(1, p, n)

# Visualização
plt.hist(data, bins=2, rwidth=0.8, align='left', color='blue', alpha=0.7)
plt.xticks([0, 1], ['Fracasso (0)', 'Sucesso (1)'])
plt.title(f"Distribuição Bernoulli (p={p})")
plt.xlabel("Resultado")
```

```python
plt.ylabel("Frequência")
plt.show()
```



Distribuição Binomial

A distribuição Binomial descreve o número de sucessos em n experimentos independentes, cada um com probabilidade de sucesso p.

A função de probabilidade é:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

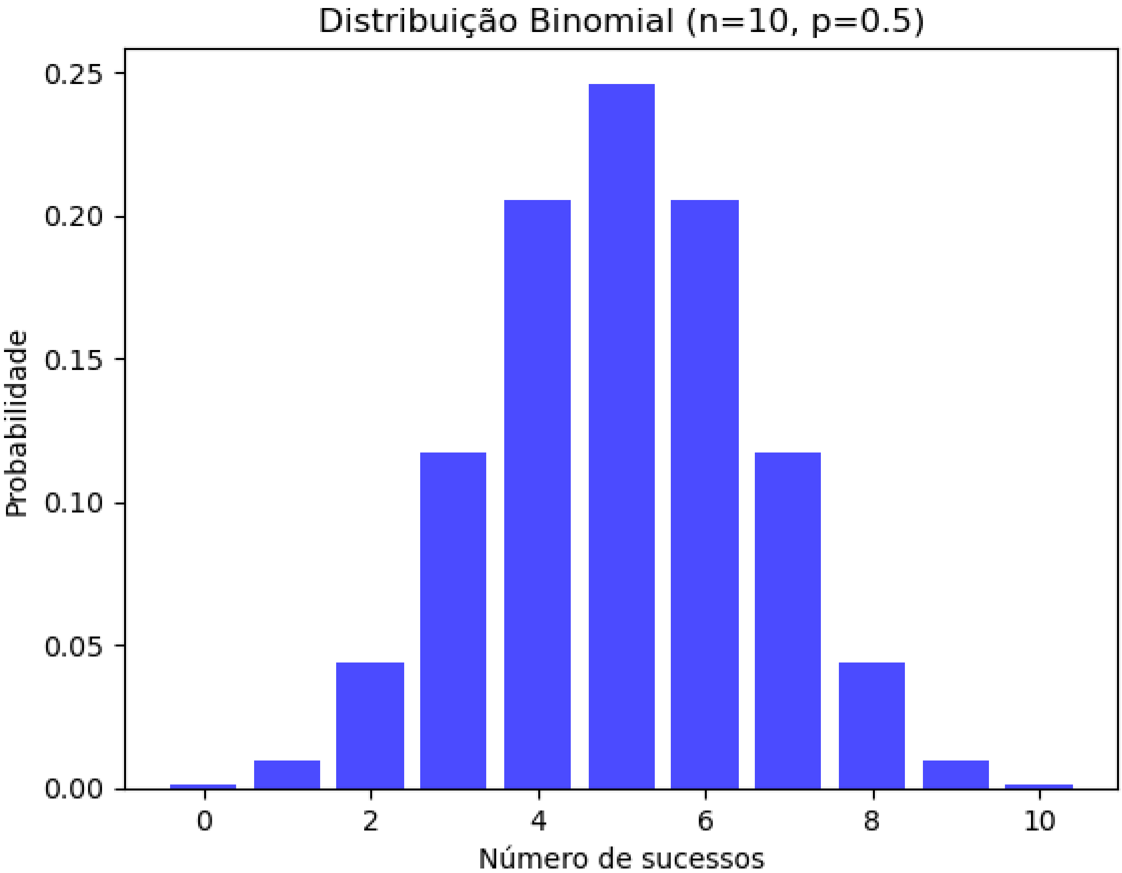
Onde:

- n: número de experimentos.
- k: número de sucessos.
- p: probabilidade de sucesso em cada experimento.

In [19]:

```python
from scipy.stats import binom

# Parâmetros
n = 10  # Número de tentativas
p = 0.5  # Probabilidade de sucesso

# Resultados possíveis
x = np.arange(0, n+1)

# PMF (Probability Mass Function)
pmf = binom.pmf(x, n, p)

# Visualização
plt.bar(x, pmf, color='blue', alpha=0.7)
plt.title(f"Distribuição Binomial (n={n}, p={p})")
plt.xlabel("Número de sucessos")
```

```python
plt.ylabel("Probabilidade")
plt.show()
```



Distribuição de Poisson

A distribuição de Poisson é usada para modelar o número de eventos que ocorrem em um intervalo fixo de tempo ou espaço, dado que os eventos ocorrem com uma taxa média conhecida λ e independentemente.

A função de probabilidade é:

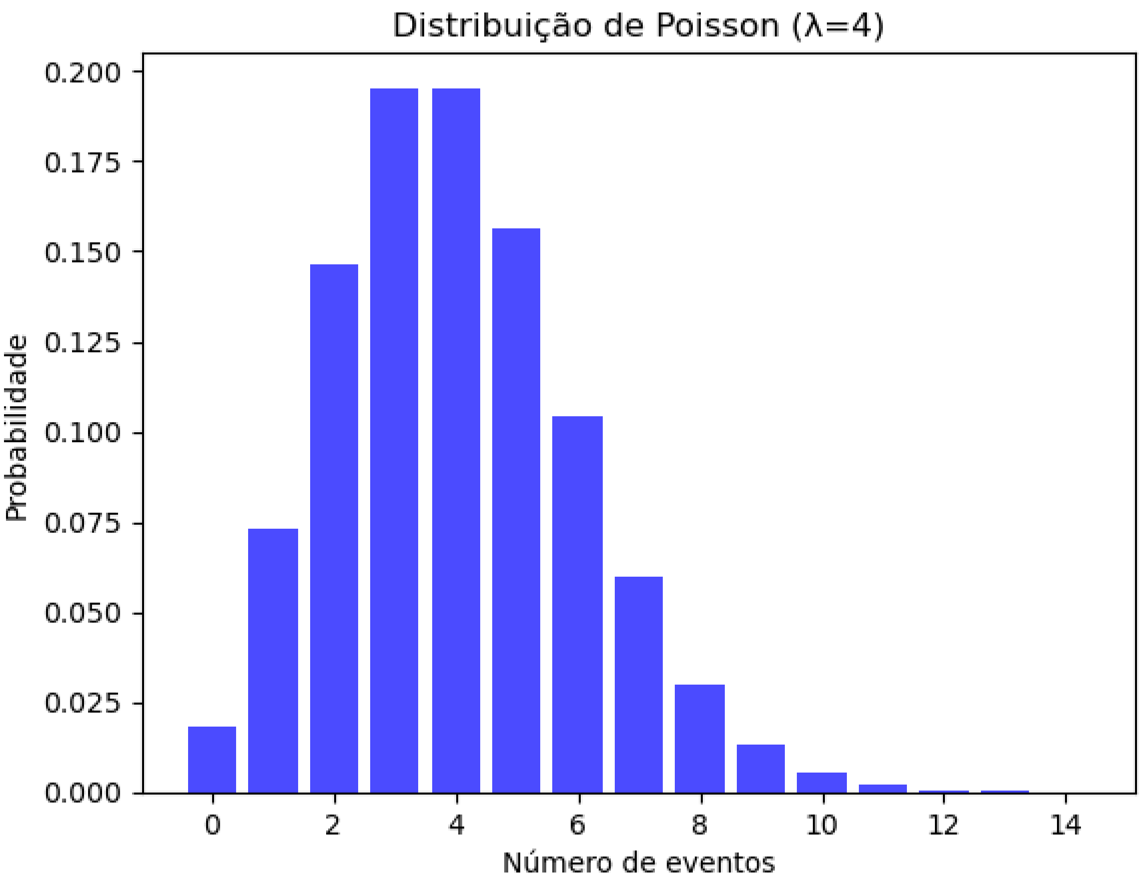
$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

Onde:

- k: número de eventos.
- λ: taxa média de eventos.

In [4]:
```python
from scipy.stats import poisson

# Parâmetro
lam = 4  # Taxa média de eventos

# Resultados possíveis
x = np.arange(0, 15)

# PMF
pmf = poisson.pmf(x, lam)

# Visualização
plt.bar(x, pmf, color='blue', alpha=0.7)
plt.title(f"Distribuição de Poisson (λ={lam})")
plt.xlabel("Número de eventos")
```

```python
plt.ylabel("Probabilidade")
plt.show()
```



Distribuição Geométrica

A distribuição geométrica modela o número de tentativas até o primeiro sucesso em uma sequência de experimentos de Bernoulli independentes.

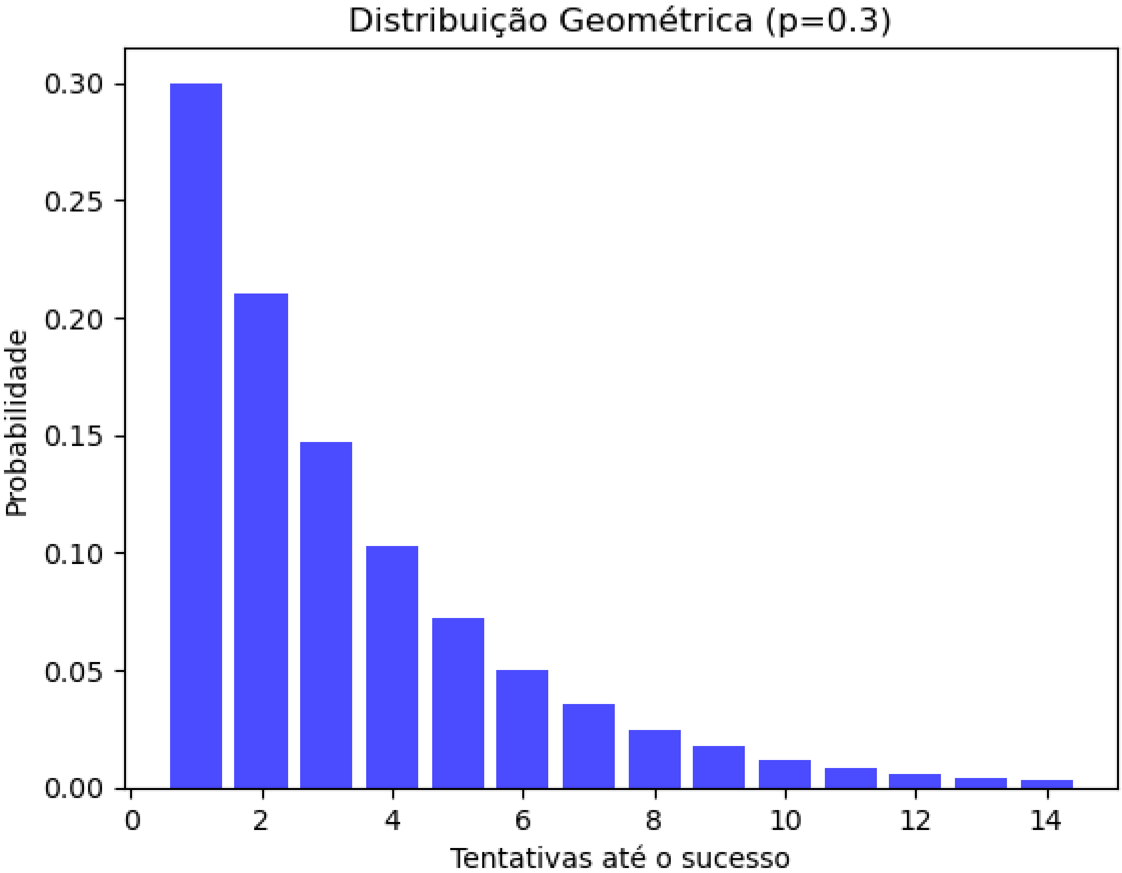
A função de probabilidade é:

$$P(X = k) = (1 - p)^{k-1} p$$

Onde:

- k: número de tentativas até o sucesso.
- p: probabilidade de sucesso em cada tentativa.

In [5]:
```python
from scipy.stats import geom

# Parâmetro
p = 0.3  # Probabilidade de sucesso

# Resultados possíveis
x = np.arange(1, 15)

# PMF
pmf = geom.pmf(x, p)

# Visualização
plt.bar(x, pmf, color='blue', alpha=0.7)
plt.title(f"Distribuição Geométrica (p={p})")
plt.xlabel("Tentativas até o sucesso")
plt.ylabel("Probabilidade")
plt.show()
```



Distribuições de variáveis aleatórias contínuas:

- Uniforme
- Exponencial
- Gaussiana (Normal)

Distribuição Uniforme

A distribuição uniforme contínua modela um intervalo $[a, b]$, onde todos os valores têm a mesma probabilidade.

A função de densidade de probabilidade é:

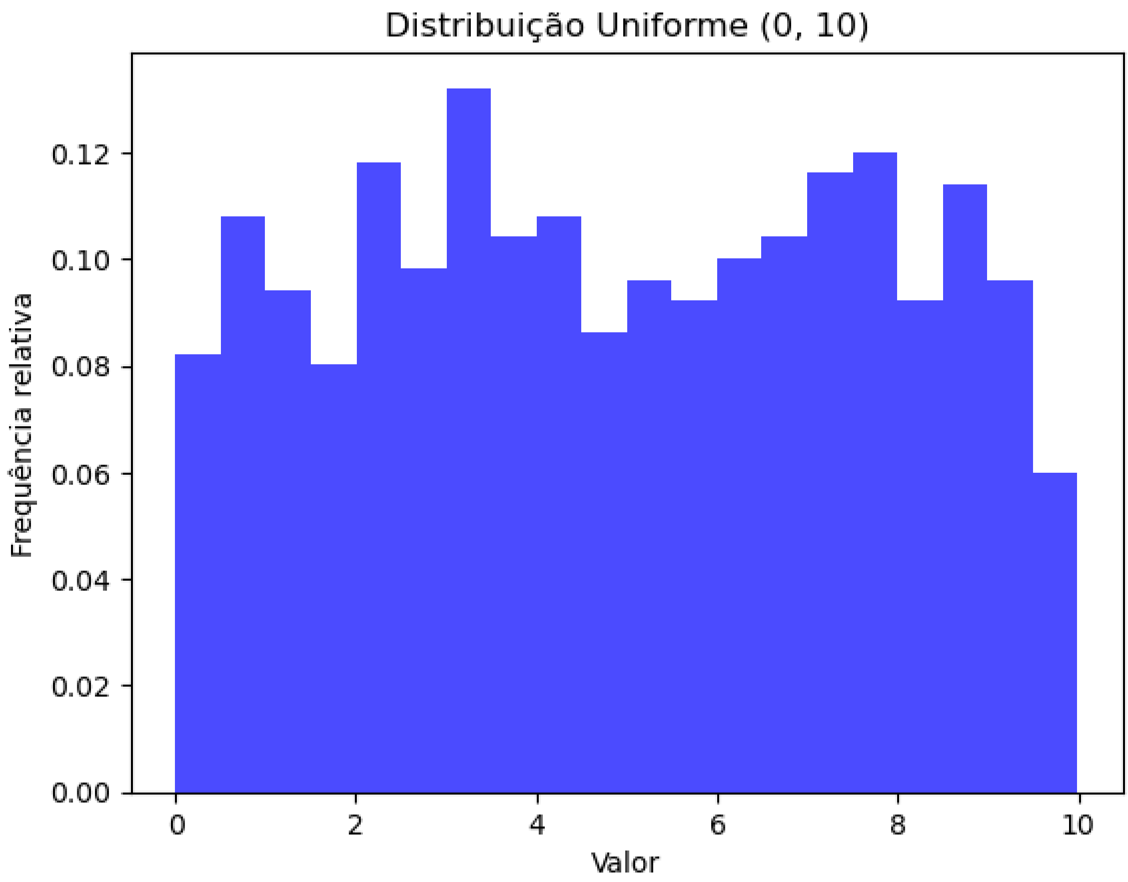
$$f(x) = \begin{cases} \frac{1}{b-a} & \text{se } a \leq x \leq b, \\ 0 & \text{caso contrário.} \end{cases}$$

Onde:

- a: limite inferior.
- b: limite superior.

In [8]:
```
from scipy.stats import uniform

# Parâmetros
a = 0  # Limite inferior
b = 10  # Limite superior

# Gerar dados
data = uniform.rvs(loc=a, scale=b-a, size=1000)

# Visualização
plt.hist(data, bins=20, color='blue', alpha=0.7, density=True)
```

```
plt.title(f"Distribuição Uniforme ({a}, {b})")
plt.xlabel("Valor")
plt.ylabel("Frequência relativa")
plt.show()
```



Distribuição Exponencial

A distribuição exponencial modela o tempo entre eventos em um processo de Poisson.

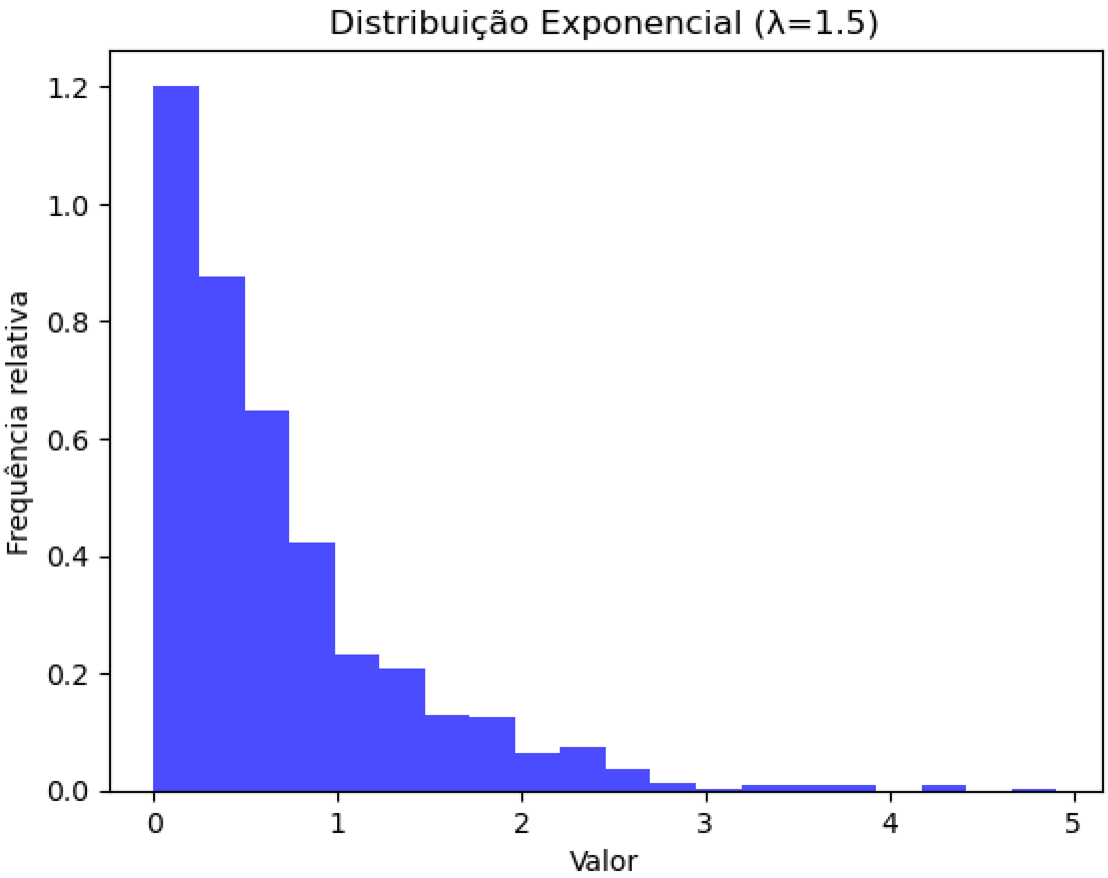
A função de densidade de probabilidade é:

$$f(x; \lambda) = \lambda e^{-\lambda x}, \text{ para } x \geq 0$$

Onde:

- λ: taxa média de eventos.

In [7]:
```
from scipy.stats import expon

# Parâmetro
lam = 1.5  # Taxa média

# Gerar dados
data = expon.rvs(scale=1/lam, size=1000)

# Visualização
plt.hist(data, bins=20, color='blue', alpha=0.7, density=True)
plt.title(f"Distribuição Exponencial (λ={lam})")
plt.xlabel("Valor")
plt.ylabel("Frequência relativa")
plt.show()
```



Distribuição Gaussiana (Normal)

A distribuição normal é a mais conhecida e amplamente utilizada em estatística, modelando muitos fenômenos naturais.

A função de densidade de probabilidade é:

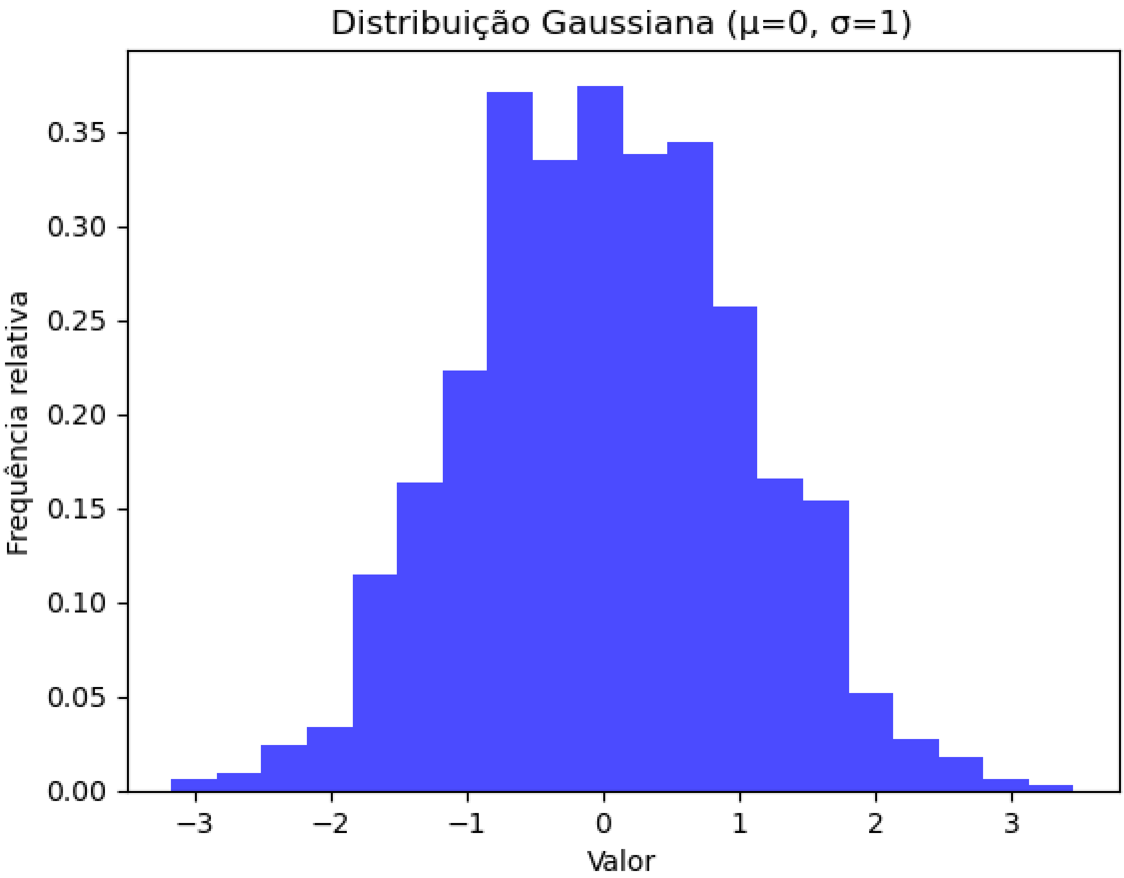
$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Onde:

- μ: média.
- σ: desvio padrão.

In [17]:

```
from scipy.stats import norm

# Parâmetros
mu = 0   # Média
sigma = 1  # Desvio padrão

# Gerar dados
data = norm.rvs(loc=mu, scale=sigma, size=1000)

# Visualização
plt.hist(data, bins=20, color='blue', alpha=0.7, density=True)
plt.title(f"Distribuição Gaussiana (μ={mu}, σ={sigma})")
plt.xlabel("Valor")
plt.ylabel("Frequência relativa")
plt.show()
```


Distribuição Gaussiana (μ=0, σ=1)
Frequência relativa
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
−3
−2
−1
0
1
2
3
Valor