

```

import numpy as np
import matplotlib.pyplot as plt

# Fonction de Runge
def runge_function(x):
    return 1 / (1 + 25*x**2)

# Génération des points de Chebyshev
def chebyshev_nodes(n, a=-1, b=1):
    k = np.arange(0, n)
    x = np.cos((2*k+1)/(2*n) * np.pi)
    return 0.5*(a+b) + 0.5*(b-a)*x

# Interpolation de Lagrange
def lagrange_interpolation(x_points, y_points, x):
    n = len(x_points)
    L = 0
    for i in range(n):
        term = y_points[i]
        for j in range(n):
            if i != j:
                term *= (x - x_points[j]) / (x_points[i] - x_points[j])
        L += term
    return L

# Étude du phénomène de Runge avec graphiques séparés
def study_runge_separate():
    ns = [5, 10, 15, 20]
    x_dense = np.linspace(-1, 1, 1000)
    f_dense = runge_function(x_dense)

    for n in ns:
        # Points équidistants
        x_eq = np.linspace(-1, 1, n)
        y_eq = runge_function(x_eq)
        y_interp_eq = np.array([lagrange_interpolation(x_eq, y_eq, xi)
                               for xi in x_dense])
        error_eq = np.abs(f_dense - y_interp_eq)

        plt.figure(figsize=(10,6))
        plt.plot(x_dense, error_eq, label=f"Erreur équidistants n={n}")
        plt.title(f"Erreur d'interpolation avec points équidistants n={n}")
        plt.xlabel("x")
        plt.ylabel("Erreur")
        plt.grid(True)
        plt.legend()
        plt.show()

        # Points de Chebyshev ♦
        x_ch = chebyshev_nodes(n)
        y_ch = runge_function(x_ch)
        y_interp_ch = np.array([lagrange_interpolation(x_ch, y_ch, xi)
                               for xi in x_dense])

```

```

error_ch = np.abs(f_dense - y_interp_ch)

plt.figure(figsize=(10,6))
plt.plot(x_dense, error_ch, '--', label=f"Erreur Chebyshev n={n}")
plt.title(f"Erreur d'interpolation avec points de Chebyshev n={n}")
plt.xlabel("x")
plt.ylabel("Erreur")
plt.grid(True)
plt.legend()
plt.show()

# Exécuter l'étude
study_runge_separate()

```



