Github: @dimonko1

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

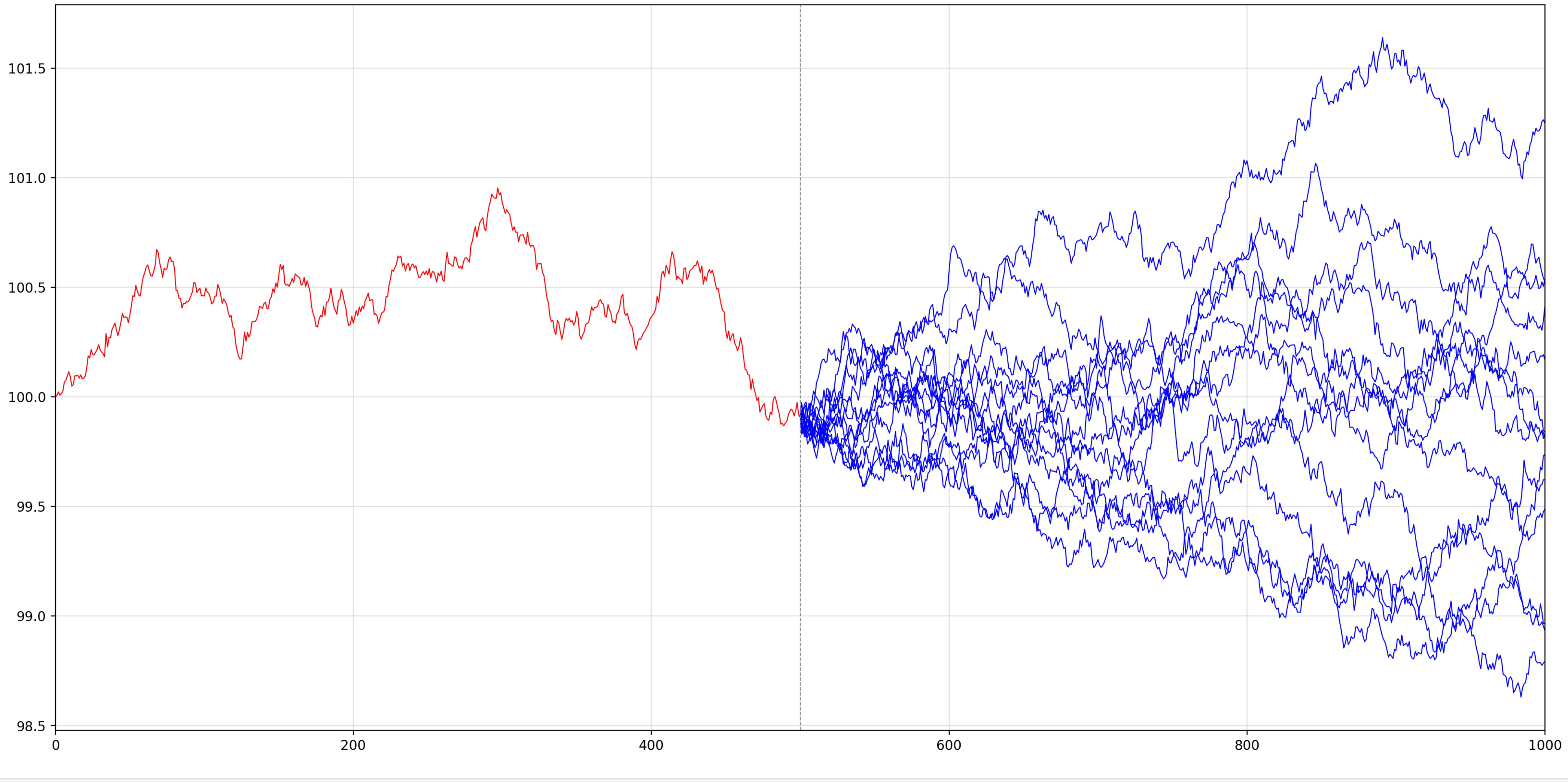
In [33]:
```
def cool_thing(n_paths, n_steps, T, mu, sigma, X_0):

    steppy = round(n_steps/2)
    X_lol = np.zeros(steppy+1)
    X_lol[0] = X_0
    dt = 0.5*T/float(steppy)
    Z = np.random.normal(0,1,steppy)
    for i in range(0,steppy):
        X_lol[i+1] = X_lol[i] + mu * dt + sigma * np.sqrt(dt) * Z[i]

    Z = np.random.normal(0, 1, [n_paths, steppy])
    X = np.zeros([n_paths, steppy + 1])
    X[:, 0] = X_lol[-1]
    for k in range(0,steppy):
        X[:, k+1] = X[:, k] + mu * dt + sigma * np.sqrt(dt) * Z[:, k]

    fig, ax = plt.subplots(figsize=(20,10), dpi=200)
    ax.grid(alpha=.35)
    ax.plot(X_lol, lw=.75, c='r')
    for i in range(1,n_paths):
        plt.plot(range(steppy, n_steps+1), X[i], lw=.75, c='b')
    ax.axvline(round(n_steps/2), c='grey', ls='--', lw=.7)
    ax.set_xlim(0,n_steps)
    ax.margins(x=0.01)
    #plt.axis('off')
    plt.show()

    return X
```

Cool vizualization of Wiener process paths simulation.

Вводные:

- n_paths – число траекторий;
- n_steps – число шагов в каждой траектории;
- T – количество лет;
- μ – mean;
- σ – variance;
- X_0 – начальная точка.

In [34]:
```
X = cool_thing(15, 1000, 1, 0, 1, 100)
```



In [ ]: