QF620 Stochastic Modelling in Finance
S1 Python Note: Brownian Motion and Martingale

We can simulate Brownian motion using the following code snippet:

```
import numpy as np
import matplotlib.pylab as plt


def simulate_Brownian_Motion(paths, steps, T):
    deltaT = T/steps
    t = np.linspace(0, T, steps+1)
    X = np.c_[np.zeros((paths, 1)),
              np.random.randn(paths, steps)]
    return t, np.cumsum(np.sqrt(deltaT) * X, axis=1)


plt.figure()
t, x = simulate_Brownian_Motion(200, 100, 10.0)
plt.plot(t, x.T)
plt.show()
```

Exercises:

1. Verify that $\mathbb{E}[W_t] = 0$ and $V[W_t] = t$.
2. Simulate the process $\alpha W_{t/\alpha^2}$.
3. Verify that $\mathbb{E}[W_t^k] = 0$ if k is an odd number.
4. Simulate the stock price process:

   $$S_t = S_0 + \mu t + \sigma W_t,$$

   where $S_0 = 50$, $\mu = 0.01$, and $\sigma = 0.2$.
5. Modify the code to simulate an n-step random walk instead, and verify that $V[S_n] = n$.