Exercise 1

Write a class `BarrierOption` extending

`net.finmath.montecarlo.assetderivativevaluation.products.AbstractAssetMonteCarloProduct`

and implementing the Monte Carlo approximation of the value of a barrier call option, which is a path dependent option with payoff

$$(X_T(\omega) - K)^+ \mathbf{1}_{\{B_L \le X_t(\omega) \le B_U \text{ for any } t \in [0,T]\}}, \quad \omega \in \Omega,$$

where $(X_t)_{t\ge 0}$ is the underlying, $K > 0$ is the strike of the option, $T > 0$ the maturity, $0 \le B_L \le B_U \le \infty$ the lower and upper barrier of the option, respectively.

Hints: to solve this exercise, it is important to start to get used with the classes and interfaces of the Finmath library which provide the simulation of stochastic processes and the valuations of financial products. In particular, since your class has to extend `AbstractAssetMonteCarloProduct`, you have to implement the method

`getValue(double evaluationTime, AssetModelMonteCarloSimulationModel model).`

To do that, you can:

- Take inspiration from the implementation of this method in

  `net.finmath.montecarlo.assetderivativevaluation.products.EuropeanOption;`

- Approximate $\mathbf{1}_{\{B_L \le X_t(\omega) \le B_U \text{ for any } t \in [0,T]\}}$ by checking that all the simulated path ω of the process until maturity satisfies this condition. You have to check which methods of

  `net.finmath.montecarlo.assetderivativevaluation.AssetModelMonteCarloSimulationModel`

  and

  `net.finmath.stochastic.RandomVariable`

  you have to use to do that.

Exercise 2

Test your implementation for an underlying following the Black-Scholes dynamics

$$dX_t = rX_t dt + \sigma X_t dW_t, \quad t \ge 0,$$

where $(W_t)_{t\ge 0}$ is a Brownian motion. In particular, consider an option with $K = 100$, $T = 3$, $B_L = 80$, B_U =`Long.MAX_VALUE`. This is basically a down-and-out option, i.e., a barrier option with only lower barrier. Its analytic value for Black-Scholes underlying is

$$BS(X_0, r, \sigma, T, K) - \left(\frac{X_0}{B_L}\right)^{-\left(\frac{2r}{\sigma^2}-1\right)} BS\left(\frac{B_L^2}{X_0}, r, \sigma, T, K\right),$$

where $BS(X_0, r, \sigma, T, K)$ is the Black-Scholes price of a European call option with initial value X_0, risk-free rate r, volatility σ, maturity T, strike K. Check how well your price approximates the analytic value taking the following parameters:

- Process parameters: $r = 0.0$, $\sigma = 0.3$, $X_0 = 100$;
- Simulation parameters: 100000 simulations, time discretization with initial time $t_0 = 0$, final time the maturity of the option, length of any time step 0.1.

Is it good? Try to make the value of the lower barrier approaching the initial value of the underlying. What do you observe?

Hints: In order to construct an object representing the simulation of a Black-Scholes model, you can use the class

`net.finmath.montecarlo.assetderivativevaluation.MonteCarloBlackScholesModel`.