Programming Monte Carlo Simulation of Stock Prices

Use `diffuse.c` in the class home page as a template, and rename it to `stock.c`.

Governing Equation: Geometric Diffusion Equation

$$dS = \mu S dt + \sigma S \varepsilon \sqrt{dt} = S\left(\underbrace{\mu dt}_{0.14\ [\mathrm{yr}^{-1}]\times\frac{1}{365}\ [\mathrm{yr}]} + \underbrace{\sigma\sqrt{dt}}_{0.2\ [\mathrm{yr}^{-1/2}]\times\sqrt{\frac{1}{365}\ [\mathrm{yr}]}}\ \varepsilon\right), \quad (1)$$

where *S* is the stock price in \$ with *dS* being the change of *S* during *dt* = 1 [day] = 1/365 = 0.00274 [yr], $\mu = 0.14\ [\mathrm{yr}^{-1}]$ is the growth rate, $\sigma = 0.2\ [\mathrm{yr}^{-1/2}]$ is the volatility, and ε is a random number following the Gaussian (normal) distribution with unit variance,

$$P(\varepsilon) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\varepsilon^2}{2}\right). \quad (2)$$

Let's precompute and save μdt and $\sigma\sqrt{dt}$ as constants.

Main Function

Reset histogram, *hist*[N_{hist} = 50] // *hist*[*i*] counts the count of ending stock prices *s* such that $i \le s < i + 1$
for *walker* = 1, N_{walker} (= 1,000)
    $S \leftarrow S_{\mathrm{init}} = \$20$
    for *day* = 1, N_{max} (= 365 days)
        $S {+}{=} S\left[\mu dt + \sigma\sqrt{dt} \times rand\_normal()\right]$
        if (*S* < 0) break
    $S \leftarrow S > 0\ ?\ S : 0$ // C notation for max(*S*, 0.0)
    ++*hist*[(int)*S*]

Box-Muller algorithm

double $rand\_normal()$
    $r_1 \leftarrow rand()/(\mathrm{double})RAND\_MAX$
    $r_2 \leftarrow rand()/(\mathrm{double})RAND\_MAX$
    return $\sqrt{-2\ln(r_1)}\cos(2\pi r_2)$ // Note the natural log function with base *e* is log() in C math library