Overview: In this lab, you will gain experience working in R while implementing a simple Monte Carlo simulation involving dice. You will also use R’s graphics capability to investigate the uncertainty inherent in estimates produced using simulation.

Implementation: In lab, together we will implement a simulation in R that simulates rolling two fair dice, allowing us to compute probability estimates for the possible sums (2, 3, …, 12). This implementation will involve:

- Writing your own functions in R, one to simulate the roll of two fair dice, one to simulate a single experiment of \( N \) rolls, and one to produce a histogram using \( M \) experiments.
- Use of the `sample()` function to simulate a fair die roll.
- Use of the `rep()` function to set up an initially zeroed vector.
- Use of the `mean()`, `sd()`, and `quantile()` functions to compute meaningful output statistics.
- Use of the `hist()` and `abline()` functions to produce an attractive display of a histogram with statistics superimposed.
- Use of the `par()` function to display multiple histograms per plot, for easy comparison.

Experimentation: After you have a working implementation, experiment with the following, considering only the probability of a sum of 7:

1. Produce a sequence of three histograms in one figure, fixing the number of estimates at 1000, but varying the number of rolls per estimate from 10 to 100 to 1000. For each histogram, superimpose the mean (in solid blue), the mean plus/minus one standard deviation (in dotted blue), and the R-default quantile values (0%, 25%, 50%, 75%, 100%; in solid red). Use the same x-axis limits across all three histograms for easy comparison. Use meaningful titles for your histograms, and save the resulting figure as a PNG for submission.

2. Produce a different sequence of three histograms in one figure, fixing the number of rolls per estimate at 1000, but varying the number of estimates from 10 to 100 to 1000. Again, superimpose the same statistics, use meaningful titles for the histograms, use the same x-limits for all three histograms, and save the resulting figure as a PNG for submission.

3. Include a plain-text README.txt in which you discuss the uncertainty produced in your figures. For example, what happens to the histogram (shape, bounds, etc.) as your parameters vary? To the mean and standard deviation? To the quantiles? Why?

4. In your README.txt, also include discussion comparing the results from the first experiment with those of the second. What differences do you notice, if any, and why?

Submitting: Package your work (`lab1.R` source, the two meaningfully named PNGs, and your README.txt) into a gzipped tarball similar to that shown below but using your netid:

```bash
tar -czvf cmsc326_lab1_bo4pz.tgz lab1/
```

presuming that your files are stored in the `lab1/` directory. Then drop your tarball into the shared Box folder for this class (do not drop into the `return/` directory there). Your lab is due by 12:00:00 (noon) on Sun 21 Jan.