İSTANBUL TECHNICAL UNIVERSITY
Department of Computer Engineering

BLG443E – Discrete Event Simulation – Spring 2019

Assignment 3 – Simplified end-to-end simulation analysis

ASSIGNMENT 3

Assignment description

Referring to the topic "Building,Validating and Using a Simulation" or the relevant chapter in the textbook, this assignment requires you to do input modelling, validation and statistical analysis with a simplified simulation. The scenario is the same as from assignments 1 and 2 and you **should use your simulator from those assignments** (preferably in a modular form) for this assignment.

If you have an alternative scenario that you wish to work with, possibly a real one, this is encouraged, but *check with the lecturer in advance.*

In your submitted notebooks, provide numbers (e.g. 1a) where you answer each question below.

Questions

1. Random Number Generation:
   a) Replace any random number generation in the simulation with your own implementation of a linear congruential generator (or a random number generator of your choice if you supply enough information – description of the algorithm, citations and references – in the report). The new generator should also be able to have a seed set for controllable replication of experiments. *Ensure the code is documented and tested in various ways in the submitted notebook.*

   ⓘ Note: You can convert cells in the notebook into "markdown" to make headings and formatted text. For example, the following text will become a title and a heading:

   ```
   This is my title
   ==========
   This is my heading
   ----------------------
   ```

2. Input Modelling:
   a) Examine the supplied data about customer arrival times (`customer_arrivals.csv`). Split the file into analysis and validation sets. *The code for doing this should be in your notebook.*
   b) Make appropriate visualisations of the analysis data. Appropriate methods for visualising such data have been shown in lectures, and are available in your class resources (textbook, slides, linked papers). *These visualisations should be included in your notebook.*
   c) Choose an appropriate probability distribution for modelling this data, or opt to use an empirical distribution. You should be able to find an appropriate distribution that has been discussed in lectures and is available in your class resources. *You should justify this choice in your notebook.*
   d) IF you are using a (parametric) distribution-based input model, calculate the parameters of

the distribution if using. IF, instead, you are using an empirical distribution, build its parameters. How this can be done has been shown in lectures and is described in the class resources. *You should show the working in your notebook, and make it clear whether you are building an empirical distribution or fitting a parametric model.*

e) Fix the simulation model to generate numbers from your chosen distribution. Wrap your random number generator from question 1. *Ensure the code is documented and tested in various ways in the submitted notebook.*

3. Input validation:

   a) Use any method as described in lectures or the class resources to validate the input model you select against validation data. *You should provide details of your approach and results in your notebook.*

4. Input/output validation:

   a) Use any appropriate method as described in lectures or the class resources to validate the full model you select against provided validation data (`experiments.csv`). *In your notebook, you should provide details of your approach and results*.

5. Experimentation:

   a) Use an appropriate method for showing outputs – do not use just point estimates – use approaches that show variance such as **interval estimates, box-whisker plots, scatter plots,** or **histograms**. *These results should be provided in your notebook along with an evaluation of these results.*

Submission policy

- Only electronic submissions through Ninova.
- No late submissions or submissions submitted otherwise than according to instructions.
- Submit:
  - A Jupyter Notebook (C++17 or Python 3.4 or later)
  - Other source files (in a **ZIP** file).

⚠ Note: as you are graded on the quality of your analysis and presentation not of your simulation code (which was the subject of previous assignments) so you should make it your code as modular as possible and keep most of the code out of the notebook itself.

- Check with the course assistants if you plan to use a framework or language not already used in this course.
- Academic dishonesty (this includes cheating, plagiarism, direct copying of code) is unacceptable.

Criteria

Criteria for a successful assignment are:

- Each part of the simulation study is carried out competently.
- Reporting is done clearly in the notebook.