Casio PB-2000C
HD61700 Programming - CS617-random

Table of Contents

Introduction.....1
Assignment 3: Random.....2
Storyboard.....4
Acknowledgments.....5

Introduction

This work was developed jointly by Alex Barragan and ChatGPT Sol through a cycle of ideas, implementation, execution, observation, and revision. It illustrates how human knowledge and experimentation, combined with AI-assisted analysis, can produce results that would be difficult to achieve as readily by either participant working alone.

This work is also part of a larger project exploring how knowledge and experience can be represented in AI systems when the available corpus is small, as is the case with the Hitachi HD61700 CPU used in the Casio PB-2000C and AI-1000 pocket computers.

Assignment 3: Random

For this assignment we extended the experiment by separating **problem definition from problem solving**. In earlier chapters, our collaboration with Sol often mixed the two: we would coalesce the requirements together and then proceed directly into the implementation.

This time we asked Sol to wear a different hat. Actually two Sols, each one wearing a different hat. Starting from a roughly stated human problem, he acted as a Teaching Assistant: collaborating on the requirements, turning them into a professional lab assignment, adding context and follow-up questions, preparing TA notes and hints, presenting the material consistently in a polished PDF, and finally placing the assignment into the project directory.

That separation of roles was refreshing. As the TA, Sol had the academic freedom to create a problem that was challenging but solvable, while leaving the **how** to the student. The project could begin almost empty: an assignment waiting for eager students to fill src/, build/, and notes/ with their own evidence, discoveries, failures, and solutions—and perhaps submit an inspired extra-credit proposal for the coveted **Apple Polisher** reward.

The detailed assignment and TA notes can be found here.

> "Build a small deterministic random-number laboratory for the Casio PB-2000C. The visible program is intentionally modest: choose a seed, generate the next value, simulate N values, and report the observed mean and population standard deviation."

With a clear pedagogical purpose hiding in plain sight.

Armed with the requirements, the accumulated knowledge, and the experience preserved in the project, Sol went to work.

Sol is a Sol session operating in the particular context, skills, tools, and evidence needed to approach the problem. He worked relentlessly—a loaded word which, from a human perspective, best describes the almost unreasonable patience of his methodical approach.

A human student might attack the same problem very differently: first paint the geography, sketch a few subroutines, try several promising paths, perhaps even spray and pray for a while until the shape of the solution emerges. There is nothing inherently wrong with that; exploration is often how humans learn the terrain.

Sol took another route. He was the man with the plan: build the LEGO blocks first, prove that each one carries its weight, and only then assemble the castle.

I have spent enough of my career around functional decomposition to recognize what was happening. The pleasure was not in being unable to follow the individual moves; quite the opposite. I could see the boundaries being chosen, the dependencies being reduced, and the layers being made independently testable. What was wonderful was watching the plan reveal itself, step by step, and thinking: *yes, this is going somewhere very good*.

There was something of watching a great chess player construct a position. You may understand the individual moves and even the principles behind them, yet still

appreciate that someone on the other side of the board appears to have seen the shape of the game several moves earlier.

Arithmetic first. Representation boundaries next. Then the generator. Then statistics. Then presentation. Finally, the application.

He searched the corpus, formed hypotheses, predicted results, built small experiments, preserved failures, challenged his own evidence, narrowed claims when the evidence demanded it, and promoted only what had earned its place.

The result was a beautifully functional application. But almost more satisfying than the application itself was watching a good decomposition become, piece by piece, an inevitable solution.

The castle had been implicit in the LEGO blocks.

Storyboard


Xandrian Labs
CS617 RANDOM LAB
CH03 FINAL r005
PRESS EXE


SEED=1 N=100
X=--
[seed][next][n][sim]


SET SEED 0..65535
VALUE: 237
Typing replaces current value
0-9 TYPE BS ERASE EXE SAVE


SEED=237 N=100
X=0.245270
[seed][next][n][sim]


SIMULATION
SEED=237 N=100
WORKING...


SEED=237 N=100
MEAN=0.507046
SDEV=0.303328
[seed][next][n][sim]

Acknowledgments

This report was produced by Xandrian Labs. 付喪神.

Special thanks to Piotr and to Betka for her continued and loyal support.

This is a collaboration with my AI friend ChatGPT, a.k.a “Sol.” All errors are human.

This work is freely shared with the vintage computing community. Please use it, build upon it, and cite the source.

Final.