A STRAND OF THE BENEDICT YEAR

The workshop

Build a working computer out of one kind of logic gate, beside the year's twelve units. At the end, the computer the child built runs a program the child wrote.

Richard Verver, JCL · Canon Lawyer

What it is

Nand to Tetris is the course by Noam Nisan and Shimon Schocken in which you build a computer from the ground up. It starts from a single chip, the NAND gate, and climbs through the adder, memory and the processor to a computer that runs programs. Part I is that hardware, in seven projects. Part II goes on to the software, all the way up to a game of Tetris, and is for a second year.

Everything the strand needs is free: the online IDE, which runs in a web browser with nothing to install and no account, and the course's own lectures on YouTube, which need no account either. The course is also sold as a Coursera class; this strand does not use it, and nothing here needs a certificate.

Why it belongs in this year

It takes apart a gap that only seems real until you look at it, the one between the machines and the spirit. A child who has built a computer from NAND gates knows that it is not magic and not a mind. It is matter given an order by a maker and kept to a rule from outside, and what it is for is in none of its parts. The map has a word for each step, so each project walks beside the units whose kind of answer it shows.

The essay Brother Electron walks the same descent, from the gate and the flip-flop to the clock and the memory, and asks what "law" can mean at each step. It is for the parent before the strand begins.

The strand at a glance

It is optional and takes about an afternoon a week beyond the year's project days: a lecture or two, something on paper, then the chip in the IDE. If it runs into the summer, let it.

PROJECT	BESIDE	WATCH
Project 0 · The workbench	Unit 1 · Reading · weeks 1–3	5 lectures, about 27 min
Project 1 · The gates	Unit 2 · Number, Unit 3 · Nature · weeks 4–9	8 lectures, about 128 min
Project 2 · The adders	Unit 4 · Making · weeks 10–12	6 lectures, about 66 min
Project 3 · Memory	Unit 5 · Memory · weeks 13–15	6 lectures, about 84 min
Project 4 · The machine's language	Unit 6 · The Person, Unit 7 · Being, Unit 8 · God · weeks 16–24	10 lectures, about 187 min
Project 5 · The computer	Unit 9 · The Act, Unit 10 · The City · weeks 25–30	6 lectures, about 95 min
Project 6 · The assembler	Unit 11 · Law · weeks 31–33	8 lectures, about 123 min
The last weeks · The machine runs	Unit 12 · Worship · weeks 34–36	—

All 49 lectures of Part I come to about 11 hours 50 minutes, or about twenty minutes a week.

Starting late

Start at Project 0 in whatever week you are in, and keep the strand's own order. Each project names the units it was written to walk beside, but no project needs its unit to have come first, and none of the units needs the workshop.

Mr. Benedict would approve

Every chip comes with a test, and the IDE says whether it passes. It cannot be talked into a pass, and nothing in it rewards recitation. The answers to every project are easy to find online, so the strand is the end-of-term trial's offered answers, every week. It is passed by not looking them up. A child who stumbles on a page of answers and says so has passed the honesty question.

The course's authors ask that solutions never be posted publicly. Keep your child's chips at home: not on a public website, a forum, or a public code repository.

The lectures

Part I has 49 lectures, most between eight and twenty minutes, about twelve hours in all: roughly twenty minutes a week. Each project page links its lectures one by one, in the order to watch them. They are the

course's own lectures, collected in a YouTube playlist by a viewer, under the Coursera course's title. Nothing there needs signing in.

Watch them together on a shared screen, with autoplay turned off. Not signed in, YouTube still shows a sidebar of other videos and a comments section; if you would rather not have those in the room, a privacy-respecting front end such as FreeTube on a computer or NewPipe on an Android phone plays the same videos without them.

The book

Nisan and Schocken's *The Elements of Computing Systems* (MIT Press, second edition 2021) is the course in print, one chapter a project. It is not needed, but a child who would rather read than watch will like it, and it makes a good gift at the end of the year.

After this year

Part II, projects 7 to 12, builds the software: a virtual machine, a compiler for a small language, an operating system, and at last a game. It needs a real programming language, so it is for a second year, and so is learning one. Project 6's lectures on writing an assembler as a program are the first look ahead.

Nand to Tetris is by Noam Nisan and Shimon Schocken, and its materials and tools are theirs, under CC BY-NC-SA 3.0. These pages link to the course and copy nothing from it; the plan, the order and the words around them are this site's.

PUBLIC DOMAIN To the extent possible under law, Richard Verver has waived all copyright and related or neighbouring rights to this page under CC0 1.0 Universal. You may copy, change, translate, print and share it, for any purpose, without asking. Quotations from other writers keep their own terms.

Read online at canon-lawyer.ca/curriculum/workshop/.