Tinker is a computer software application for molecular dynamics simulation with a complete and general package for molecular mechanics and molecular dynamics, with some special features for biopolymers. The core of the package is a modular set of callable routines which allow manipulating coordinates and evaluating potential energy and derivatives via straightforward means.

Tinker works on Windows, OS X, Linux and Unix. The source code is available free of charge under a restrictive license. The code is written in portable FORTRAN 77, FORTRAN 95 or CUDA with common extensions, and some C.

**Core developers are:**

1) the Jay Ponder lab, at the Department of Chemistry, Washington University in St. Louis, St. Louis, Missouri. Laboratory head Ponder is Full Professor of Chemistry (main appointment), Biochemistry and Molecular Biophysics, and Biomedical Engineering

2) the Pengyu Ren lab , at the Department of Biomedical Engineering University of Texas in Austin, Austin, Texas. Laboratory head Ren is Full Professor of Biomedical Engineering

3) Jean-Philip Piquemal's research team at Laboratoire de Chimie Théorique, Department of Chemistry, Sorbonne University, Paris. Research team head Piquemal is Full Professor of Theoretical Chemistry.

**The Tinker package is based on 3 codes:**

1. The canonical Tinker (version 8.)
2. The Tinker-OpenMM package for Tinker's use with GPus;
3. The Tinker-HP package for massively parallel MPI applications on CPus and GPUs . Tinker-HP received the 2018 Atos-Joseph Fourier Prize in High Performance Computing.

**Programs are provided to perform many functions including:**

1. Energy minimizing over Cartesian coordinates, torsional angles, or rigid bodies via conjugate gradient, variable metric or a truncated Newton method
2. Molecular, stochastic, and rigid body dynamics with periodic boundaries and control of temperature and pressure
3. Normal mode vibrational analysis
4. Distance geometry including an efficient random pairwise metrization
5. Building protein and nucleic acid structures from sequence
6. Simulated annealing with various cooling protocols
7. Analysis and breakdown of single point potential energies
8. Verification of analytical derivatives of standard and user defined potentials
9. Location of a transition state between two minima
10. Full energy surface search via a Conformation Scanning method
11. Free energy calculations via free energy perturbation or weighted histogram analysis
12. Fitting of intermolecular potential parameters to structural and thermodynamic data
13. Global optimizing via energy surface smoothing, including a Potential Smoothing and Search (PSS) method

**Here are six reasons to encourage your child’s love of tinkering:**

1. **Building hobbies builds confidence.** When your child has a chance to activate their creativity, their self-confidence grows. Exploring an interest, especially one as creativity-driven as tinkering with robots or drones, gives your child a sense of purpose and accomplishment.
2. **They’ll fail—and that’s okay.** For every successful invention, there are dozens of failed prototypes. Failure builds determination, and chances are that you learned this lesson as a kid (how many mud pies did you make that just wouldn’t hold their shape?). Encouraging your child to tinker is a great way to pass on the lesson that success requires failure.
3. **Hands-on learning is powerful.** The Maker Movement and project-based learning have spread to classrooms, and we’re realizing that children thrive when they have the opportunity to learn by doing. Let your child take their education into their own hands (literally), and see where it takes them.
4. **They’ll see that learning equal to fun**. We know that tinkering builds young brains, and messing around with robots and drones is undeniably fun. Let your child build the connection between the two. When your child categorizes learning as enjoyable, they’re setting themselves up for success throughout their education and career.
5. **Coding is a timeless skill.** From Ada Lovelace to Bill Gates, coders have made names for themselves in recent history. And now more than ever, programming is a basic literacy. While technology does evolve, what remains consistent is the analytical imagination that coding requires, and connected toys lay the groundwork for this kind of thinking and exploration.
6. **They may just change the world.** We’re not saying that every child who tinkers will go on to develop life-changing inventions…but what if they do? Give your child the tools and agency to change their world.