Problems

On the topic of the general idea of problem solving:

(Why this split into traditionally and now(this actually started as a paragraph and then I realized it might come through clearer as a table)? To make the point that now we have all these tools out there, but they can someone be more confusing than the traditional approach, requiring a fair amount of original effort to get onto using them)

|  |  |
| --- | --- |
| Traditionally | Now |
| Only infrastructure you could rely on was paper and pen. | Not so much, we use computers for most parts of problem solving |
| (Side-note: hence tools were the same for everyone being smart, doing calculations quickly in one’s mind, or being fast at algebra was really what mattered). |  |
| Effortless to learn how to solve problems, just apply all the “simple algorithms” learn in school, for simplifying equations, graphing data, and so on. This can all be done in the same workspace | “Programming” has become the new tool. We all know how powerful this is, but why would it be a problem?   * In programming you type strings(either into a file, or a command line) most of the time, then execute, and eventually you get an output. * Yes, lots of software tools don’t rely on string inputs, but rather have graphical user interfaces, we will discuss the problem with this later on. |

So you say, but, there are user interfaces to help out for the people that can’t or won’t learn how to program. And yes, you are right, but here is the difference between the pen and paper, and using a program. A person who knew how to use a pen and paper, and realized that solving some problem graphically(e.g. graph intersection) was a good tool for his problem can use his tools(pen and paper) to generate this other tool(a “graphical solver”).

Meanwhile, say you had some software tool. Let’s pick Solidworks, which is an excellent example of a software package that can very well rely on its graphical user interface for doing what the user desires: geometrical design (and perhaps some physical analysis, or getting drawings for manufacturing). Now this is way beyond what you could ever do by hand, obviously, but think you somehow wanted to edit slightly some functionality in Solidworks, perhaps you have discovered a way you could boost your productivity by having shortcuts for certain actions, etc… well, with advanced software packages as Solidworks, that might be a built in feature (the ability to record macros perhaps?). But the more complicated your add-ins need to be you realize you need to program. So we are back at our initial problem: writing strings to a file or command console.

* Another important point with engineering software is that a lot of times you have software that can do something very specifically and nothings else. And on top of that it is a black box: you enter the parameters and out comes a solution.

Ideas: some ideas spur other as we are developing (analog to being on Wikipedia and clicking on one link after another), and might result in a great final idea, but for someone from the outside it’s impossible to track the origin and understand where and how the idea came to be. If someone else went in and looked at the origins, perhaps they would be lead down a different path with interesting ideas?