

Well, Then

- we now know that there are well-defined mathematical problems that can't be solved by any algorithm whatsoever
 - for example, there's no algorithm that can decide, for every program and input, whether the program halts on that input
- this isn't a dead end!

Well, Then

- we got here by building up a collection of tools and concepts
 - *logic, deduction, and proofs* – to construct rigorous arguments and establish truth
 - *functions, sets, and counting* – building blocks
 - *functions* – computation is mapping inputs to outputs
 - *sets* – fundamental objects (languages are sets)
 - *counting* – distinguishing between sets that can be systematically enumerated and those that cannot
 - *formal languages* – to frame problems in terms of which strings can be generated and recognized
 - an aside: practical applications of regular expressions and context-free grammars
 - *models of computation* – to formalize what it means to compute
 - *logic circuits* – to show how computation can be built from simple mechanical components
 - *finite state automata (DFAs, NFAs, PDAs)* – to connect language recognition with concrete computational processes
 - *general grammars* – to extend the expressive power of language descriptions toward general computation
 - *Turing machines* – to capture the full power of algorithmic computation and reveal its fundamental limits

Where To Now?

these areas all build on the same foundations: understanding what computation is, what it can do, what it can do efficiently, and where its limits lie

- algorithms and complexity
 - how do we design algorithms?
 - how do we make them efficient?
 - what makes problems tractable vs intractable?
 - what can we do when exact solutions are impractical?
- systems and applications
 - how do we build machines to carry out computation? (*computer architecture*)
 - how do we design and implement languages for expressing computation? (*programming languages, compilers*)
 - how do we use build systems that use computation to solve real-world problems? (*software development*)
 - how do we carry out many computations at once efficiently and safely? (*OS, systems*)
 - what can we do with large amounts of data? (*machine learning / AI*)