

Theory: Intro to computational thinking

⌚ 4 minutes

2 / 3 problems solved

Start practicing

12482 users solved this topic. Latest completion was about 9 hours ago.

A programming language allows us to explain to a computer how to execute a solution to a particular problem in the real world. But first, a programmer has to come up with the solutions, also using a skill necessary for any type of programming no matter the language. It is called computational thinking.

§1. What is computational thinking?

Computers can only deal with clear, concise instructions that agree with the rules of formal logic. However, real-world problems are rarely cut so clearly. **Computational thinking** is a set of mental skills helping to see the problems as a set of complex information processes that we can transform into a particular set of instructions for a computer.

Approach every problem you encounter while learning a programming language as not only the opportunity to remember the syntax of the language but also as an opportunity to train computational thinking.

To do that, you can follow a simple algorithm:

1. Describe the problem
- What exactly needs to be done? What input data are you given and what does the desired outcome look like?
2. Identify the important details needed to solve this problem
- Before thinking of a solution, make sure you took into account all the important aspects of the problem. The devil is in the details, and in case of programming, it hides in edge cases.
3. Decompose
- Break the problem down into small, logical steps until you know exactly how to code each part of it.
4. Use these steps to create an algorithm that solves the problem
- Connect the pieces of the problem in a way that would produce the desired outcome in all specified cases.
5. Evaluate the process
- Usually, a problem has at least a few solutions, and it's very useful to evaluate your idea to make sure you've chosen a way that is as efficient as possible.

§2. Conclusion

Since computational thinking is a skill, it requires lots of practice before you can easily apply it. Don't despair if your only thought when looking at a problem is that you have no idea how to achieve the desired results. Use the algorithms we described and keep on breaking the problem down until you see how to explain it to a computer with the tools of the programming language you chose.

📄 Report a typo

986 users liked this theory. 10 didn't like it. What about you?



Start practicing

Current topic:

✓

Intro to computational thinking.

Z★ ...

Topic is required for:

✓

Components of computational thinking.

Z★ ...

Table of contents:

1 Intro to computational thinking

§1. What is computational thinking?

§2. Conclusion

Feedback & Comments

Comments (7)

Hints (0)

Useful links (0)

Show discussion

