

LESSON 12

Heuristics

40 minutes

Foundations of Computer Science

Outcome 1.10

Outcome 1.2

Outcome 1.5

Outcome 1.6

Outcome 1.8

Outcome 1.14

Outcome 1.11

WHAT THIS LESSON DOES

In this lesson, you'll explore problem-solving strategies that prioritise speed over perfection. Learn what heuristics are, how they work with practical examples, their limitations, and applications in computing and daily life. Engage in a group discussion to apply your understanding.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
1.10	discuss when heuristics should and could be used and explain the limitations of using heuristics
1.2	explain how the power of computing enables different solutions to difficult problems
1.5	evaluate alternative solutions to computational problems
1.6	explain the operation of a variety of algorithms
1.8	evaluate the costs and benefits of the use of computing technology in automating processes
1.14	explain when and what machine learning and AI algorithms might be used in certain contexts
1.11	discuss the complex relationship between computing technologies and society including issues of ethics

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand the concept and purpose of heuristics in problem-solving.
- Recognise how heuristics function through practical examples.
- Identify the limitations and potential drawbacks of using heuristics.
- Explore real-world and computing applications of heuristics.
- Engage in collaborative discussion to apply heuristic principles.

BEFORE THE LESSON

- No environment or hardware needed. This is a discussion and theory lesson, so the main prep is having a concrete worked example ready: the travelling salesman nearest-city rule and a spam filter are both mentioned and both work well on the board.
- Decide group sizes and have one or two discussion problems chosen in advance so the final activity starts cleanly rather than stalling on what to discuss.

Heuristics

HOW THE LESSON RUNS

Introduction

What are Heuristics?

How Heuristics Work

Limitations of Heuristics

Applications of Heuristics

Group Discussion Activity

Wrap Up

TEACHING MOVES

- 2. What are Heuristics? Pin the core contrast early: an algorithm guarantees the best answer, a heuristic trades that guarantee for speed. Frame it as a rule of thumb they already use, like estimating a shopping total by rounding.
- 3. How Heuristics Work. Walk the nearest-city greedy example step by step on the board so they see a heuristic making one local choice at a time. Stress it never looks at the whole map, that is the point and also the weakness.
- 4. Limitations of Heuristics. Return to the greedy route and show a case where always picking the nearest city gives a worse total than a longer first hop. Concrete beats the abstract word suboptimal.
- 5. Applications of Heuristics. Ask which of their everyday apps could not run if they had to compute the perfect answer every time. Route planners and spam filters make the necessity of good-enough obvious.
- 6. Group Discussion Activity. Hold each group to the four prompts in order: what makes it complex, propose a rule, name what could go wrong, where else it applies. Naming the limitation is the part they will skip, so ask for it directly.

WHAT TO WATCH FOR

- Students treat heuristic and algorithm as interchangeable, or think a heuristic is just a worse algorithm. Fix on the guarantee: an algorithm promises the optimal answer, a heuristic promises only a quick reasonable one. The heuristic is often the right choice, not the inferior one.
- Students assume a heuristic that works well on one problem transfers to any problem. Point to the problem-specific adaptation limitation. Ask what happens if you use nearest-city logic to filter spam. The mismatch makes the point that heuristics are tailored, not universal.

DIFFERENTIATION

- Emerging: Give a single sentence template to start their discussion: My shortcut is to always... which could go wrong when... Anchor them to one worked example, the delivery route, rather than the full list of applications.
- Developing: Ask them to explain aloud why their proposed rule saves time before the class moves on, so they connect the shortcut to the cost it avoids.