

Week 1 Seminar: What Is AI?

AI or not AI, four puzzles, and AI around us

This seminar uses **no computers**. Work in groups of 3 or 4 with paper and a pen. It is **not assessed**. The four puzzles in Section B preview methods you will meet later in the module: search (Week 2), logic (Week 3), planning (Week 4) and constraint satisfaction (Week 6). For each puzzle, **how** you solved it matters more than the answer.

Timing: about 2 hours 15 minutes, with a 10-minute break after Activity 3.

A AI or not AI?

Activity 1 | Vote

15 minutes

For each item, decide as a group: is it AI, not AI, or are you not sure?

Item	AI	Not AI	Not sure	Item	AI	Not AI	Not sure
A spam filter	☐	☐	☐	Netflix recommendations	☐	☐	☐
A thermostat	☐	☐	☐	A chess app	☐	☐	☐
A sat-nav finding a route	☐	☐	☐	A calculator	☐	☐	☐
A spreadsheet formula	☐	☐	☐	Face unlock on a phone	☐	☐	☐
ChatGPT	☐	☐	☐	Automatic doors	☐	☐	☐

- What rule did your group use to decide?
- The lecture defined an *agent* as an entity that perceives its environment and acts upon that environment. Is a thermostat an agent? Is it intelligent?
- Does it matter what we call "AI"? Who gains from calling a product AI?

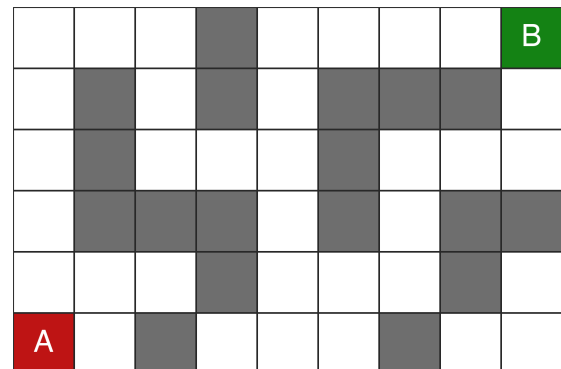
B Four puzzles

Activity 2 | The maze

15 minutes

Find your way from A (red) to B (green). You can move up, down, left or right, but not through the grey squares.

- (a) Find a path from A to B. How many steps is it?
- (b) Write pseudocode that finds a path through *any* maze, even one you have never seen. Assume your program can see the whole maze.
- (c) Does your pseudocode always find a path if there is one? Does it always find the shortest?
- (d) What does your program need to remember as it goes?



Activity 3 | The river crossing

15 minutes

A farmer must take a wolf, a goat and a cabbage across a river. The boat carries the farmer and at most one other. If the wolf and the goat are left alone together, the wolf eats the goat. If the goat and the cabbage are left alone together, the goat eats the cabbage.

- (a) Find a way to get all four across safely.
- (b) Write down every state as *left bank* | *right bank*. The start is F W G C | and the goal is | F W G C.
- (c) Which states are not allowed?
- (d) What is the smallest number of crossings?

Activity 4 | Knights and knaves

15 minutes

On an island, every person is either a *knight* or a *knave*. Knights always tell the truth. Knaves always lie.

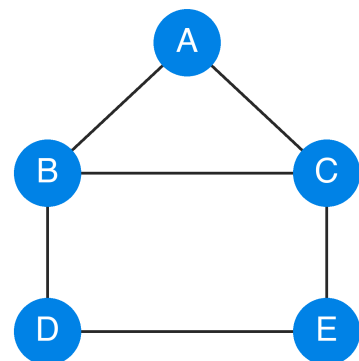
- (a) A says: "I am both a knight and a knave." What is A?
- (b) A says: "We are both knaves." B says nothing. What are A and B?
- (c) Write your reasoning for (b) as "if ... then ..." rules.
- (d) *Harder*. A says: "We are the same kind." B says: "We are of different kinds." What are A and B?

Activity 5 | The exam timetable

15 minutes

Five exams, A to E, must fit into three days: Monday, Tuesday and Wednesday. A line between two exams means they share students, so they cannot be on the same day.

- (a) Find a timetable that works.
- (b) Describe your method, step by step.
- (c) Did you ever have to undo a choice? When?
- (d) Wednesday is cancelled. Can the exams fit into two days? Why or why not?



Exam	A	B	C	D	E
Day					

Activity 6 | What did you just do?

10 minutes

- (a) Would your maze method still work on a maze with a million squares? Would your timetable method work for 500 exams?
- (b) In each puzzle, what did you have to keep track of?

C AI around us

Activity 7 | AI in an organisation

25 minutes

Choose one organisation: a supermarket, a bank, the NHS or a university.

- (a) Where does it use AI, or where could it? Find three uses.
- (b) For each use, give one benefit and one risk, and say who is affected.
- (c) Choose your best example to share with the class. Which of the module's eight topics do you think it uses?

Use of AI	Benefit	Risk	Who is affected?

D Before Week 2

Setup checklist

- ☐ Install Python 3 from python.org (or check it is installed: `python3 --version`).
- ☐ Download the module code from github.com/sakibanwar/comp5001-ai-code (green **Code** button, then **Download ZIP**) and unzip it.
- ☐ In a terminal, go into the `week02-search` folder and run `pip install -r requirements.txt`.
- ☐ Run `python maze/maze.py maze/maze1.txt` and open the picture `maze.png` it makes.

If something goes wrong, bring the error message to the Week 2 practical.