C++ Foundations for Solving Global Challenges

Dr Stelios Sotiriadis

Who am I?

1. PROFILE

A. Professor of CS at Birkbeck, and CEO of Warestack.

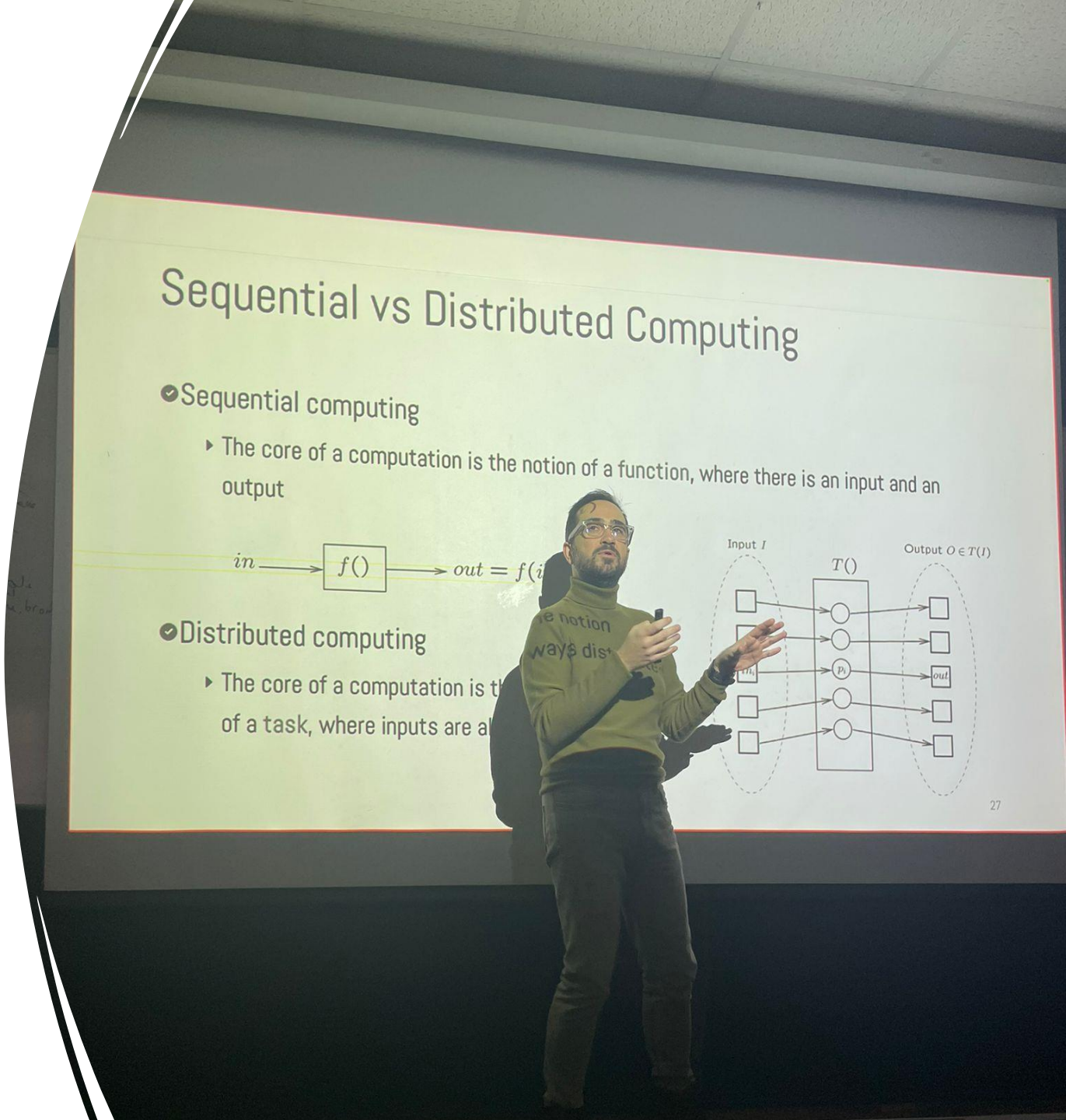
Previously at UofT, visiting Prof at ND and Boston University

2. TEACHING

Computer Science, Data Engineering, and Applied Artificial Intelligence. Prog. Director of MSc Data Science & AI

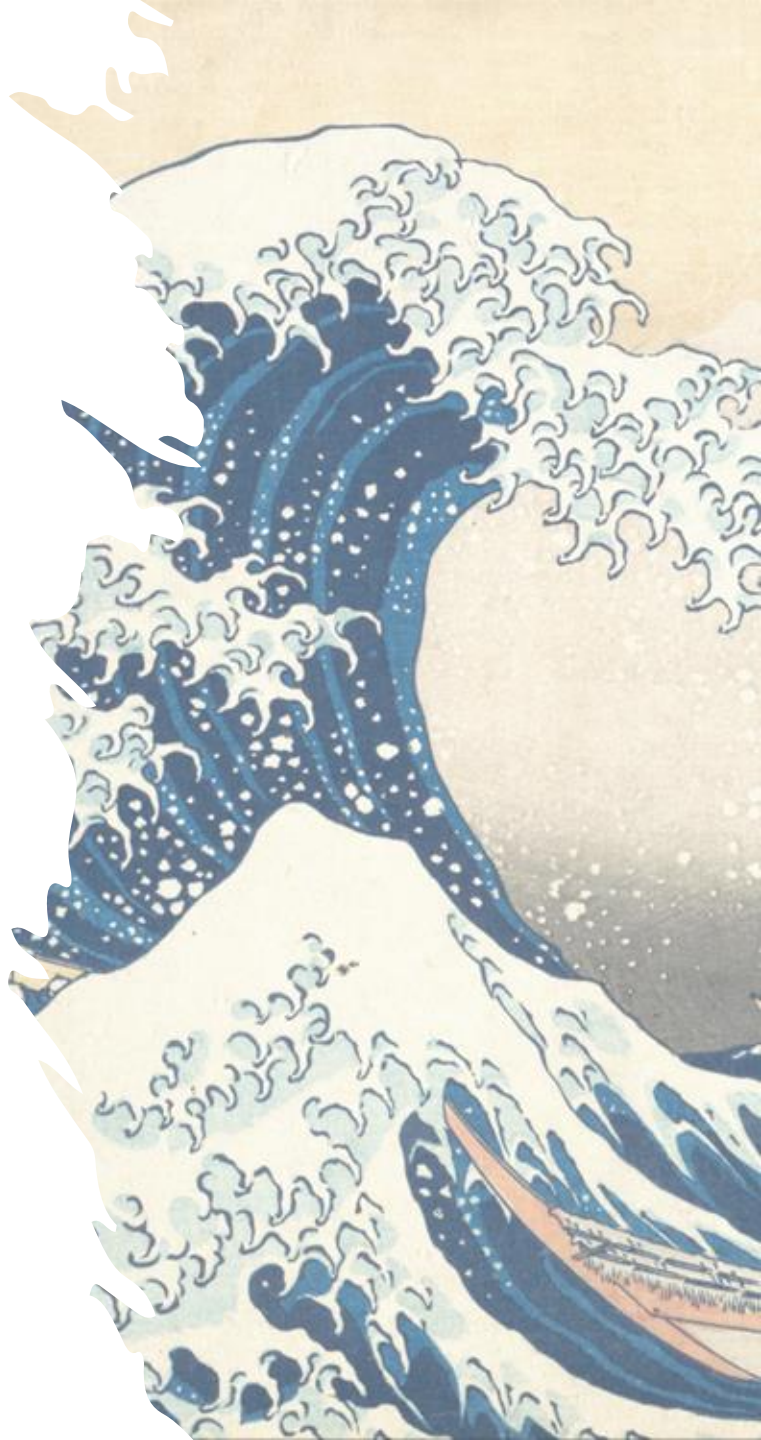
3. RESEARCH

Make computer systems and developers work faster and better. Projects with IBM, Huawei, Autodesk and startups



Learning outcomes

1. Explain what a C++ program is
2. Use variables and input/output
3. Apply conditionals and loops to make decisions
4. Understand how code processes real-world data
5. See how software helps address global challenges



Lesson Roadmap

1. Programming and Global Challenges (5 min)
2. C++ foundations (15 min)
3. Applied global challenge example (15 min)
4. Interactive exercise (10 min)
5. Reflection and Q&A (5 min)

Programming and Global Challenges

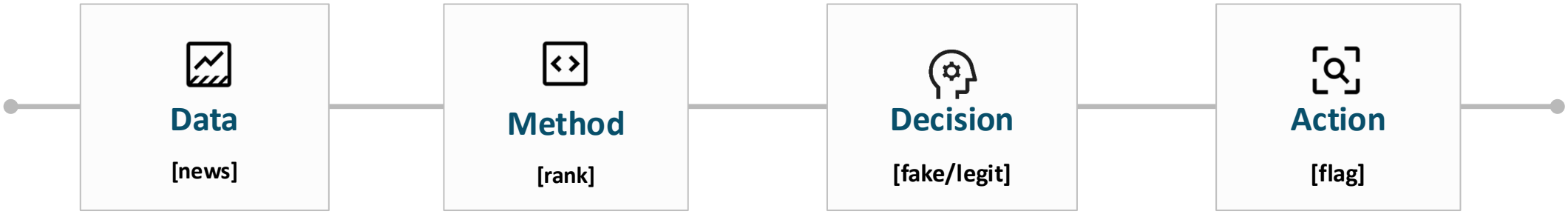
PART 1

Interdisciplinary use case

Fake news are spreading online.

How could we verify if information is real?

How could code help with classifying fake content?



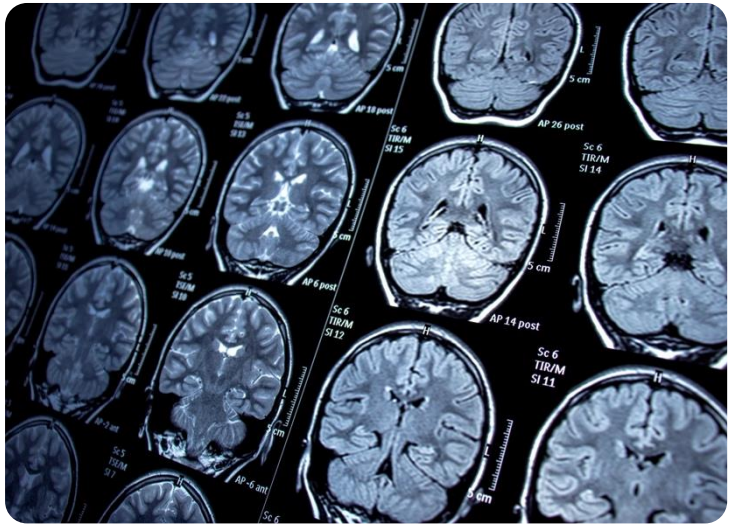
We'll build a small version of this analysis pipeline today.

Why learn C++ as a non-CS students?

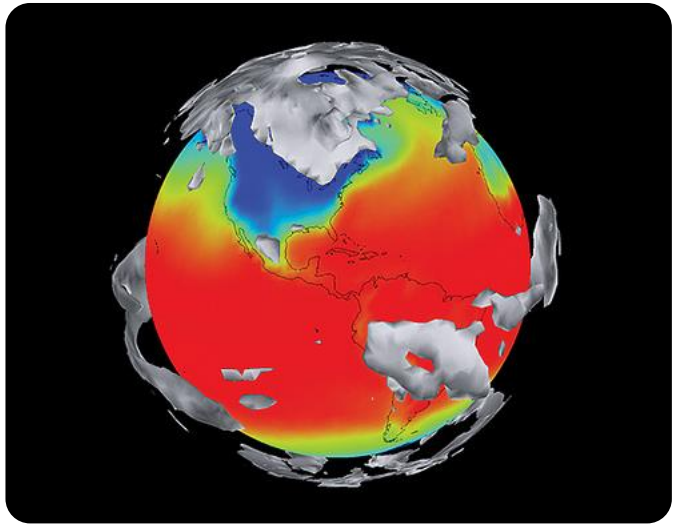
- Programming = modern literacy
- Data exists in every field
- C++ helps build tools for problem solving
- Real world data analysis and processing foundations

C++ is powering global systems

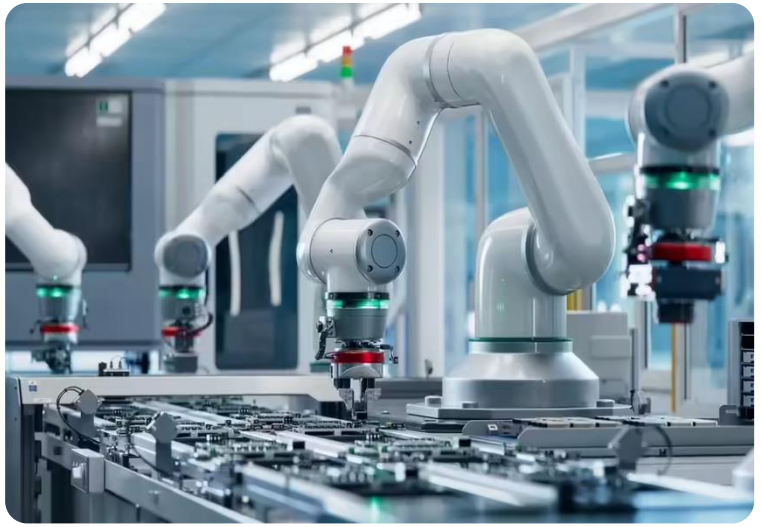
MEDICAL IMAGING



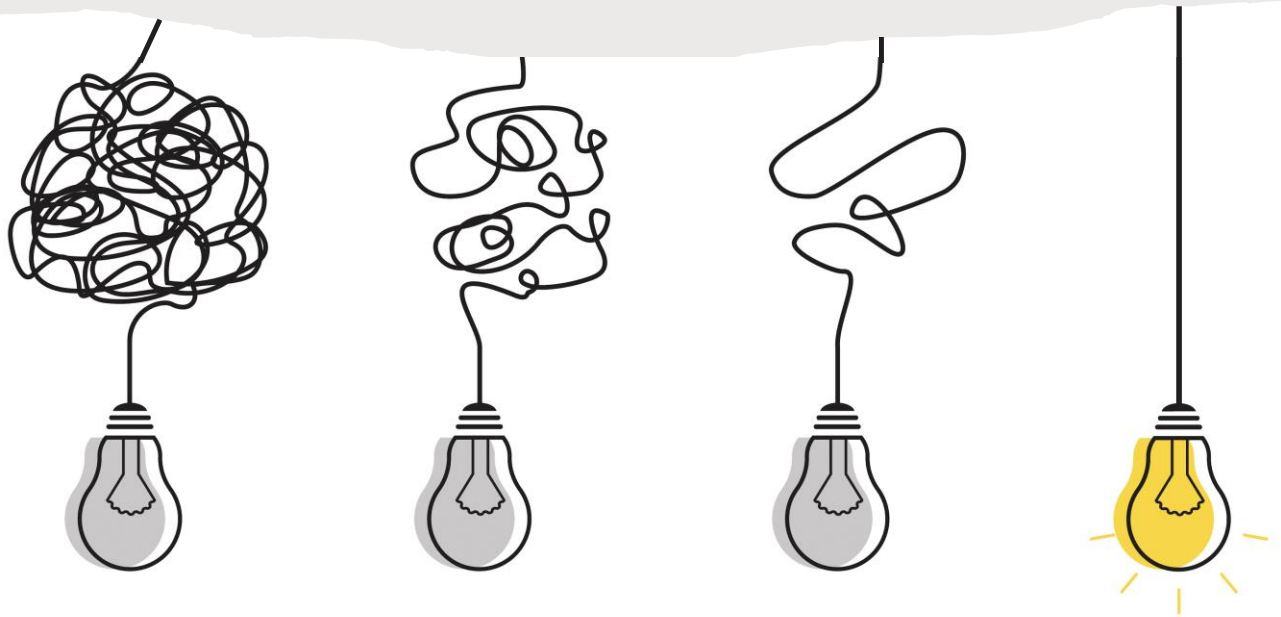
CLIMATE MODELING



AUTONOMOUS SYSTEMS



Programming = Problem Solving



Programming is not only about writing code; it's about systematic thinking.

C++ foundations

PART 2

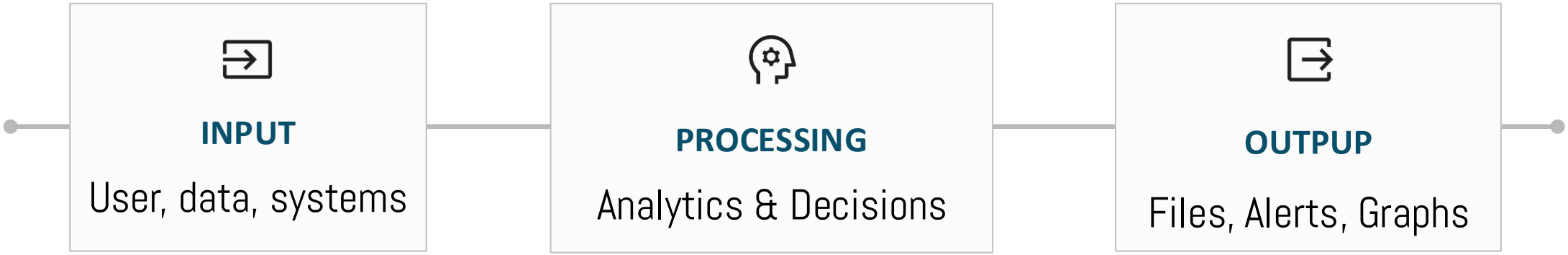
What is C++?

- A **programming language** used to build software:
  - Runs very fast & used for high-performance systems
  - Solves real-world problems
  - Powers operating systems, games, real-time systems, and AI

Is chosen when performance and reliability matter.

What is a program?

A set of instructions designed to process information and solve a problem.



What is the INPUT, PROCESSING and OUTPUT?

Given a dataset of news articles, determine which ones are most likely fake or AI-generated

INPUT

Dataset of news articles

PROCESSING

Analyze and classify each article content

OUTPUP

Articles most likely to be AI-generated/fake

Your First C++ Program

```
#include <iostream>


int main() {
    std::cout << "Fake News!";
    return 0;
}
```

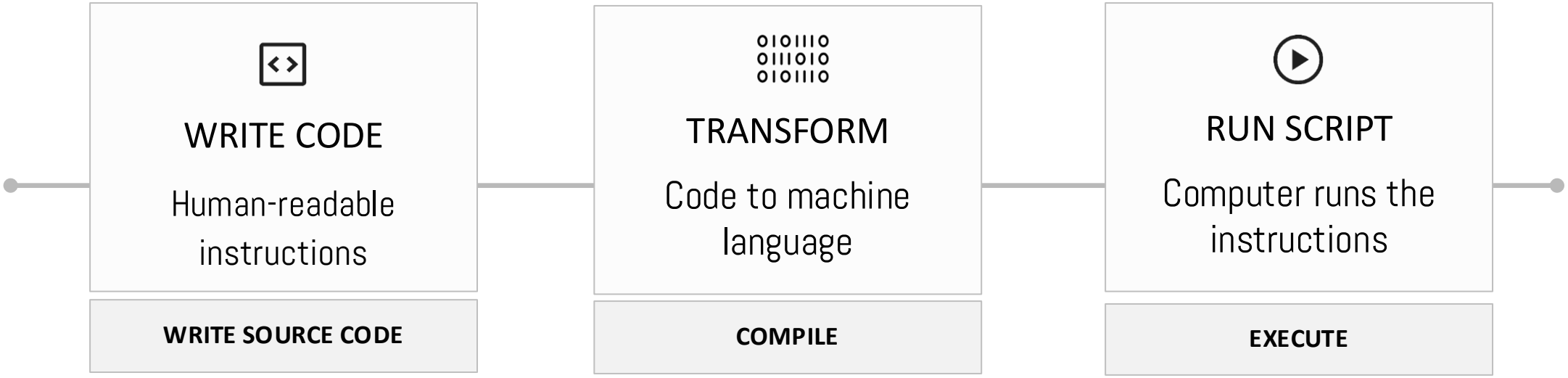
Tells C++ to include the input/output library

Every C++ program starts running from `main`

Prints `Fake News!` in the screen

This ends the program

The Path from Code to Action



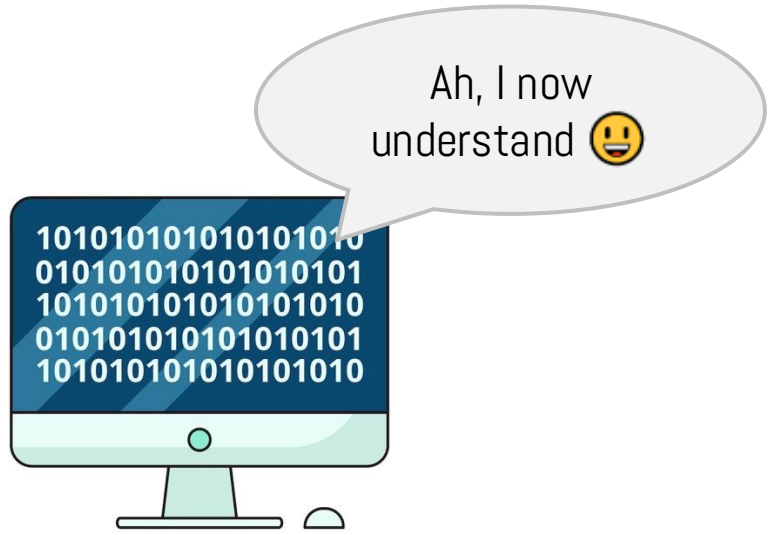
C++ Compiler

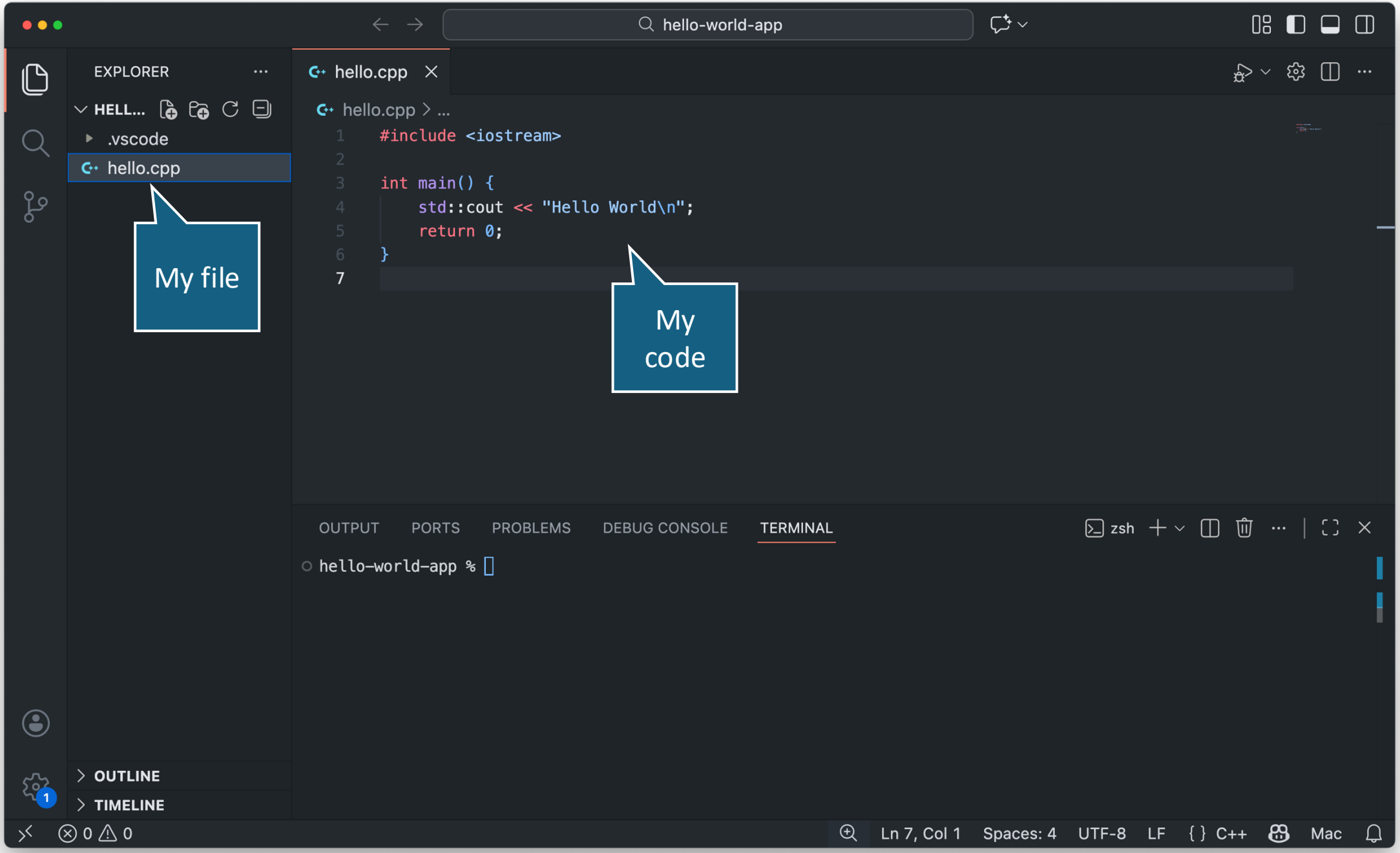
A C++ compiler is a program that translates your human-written code into machine code so the computer can run it.

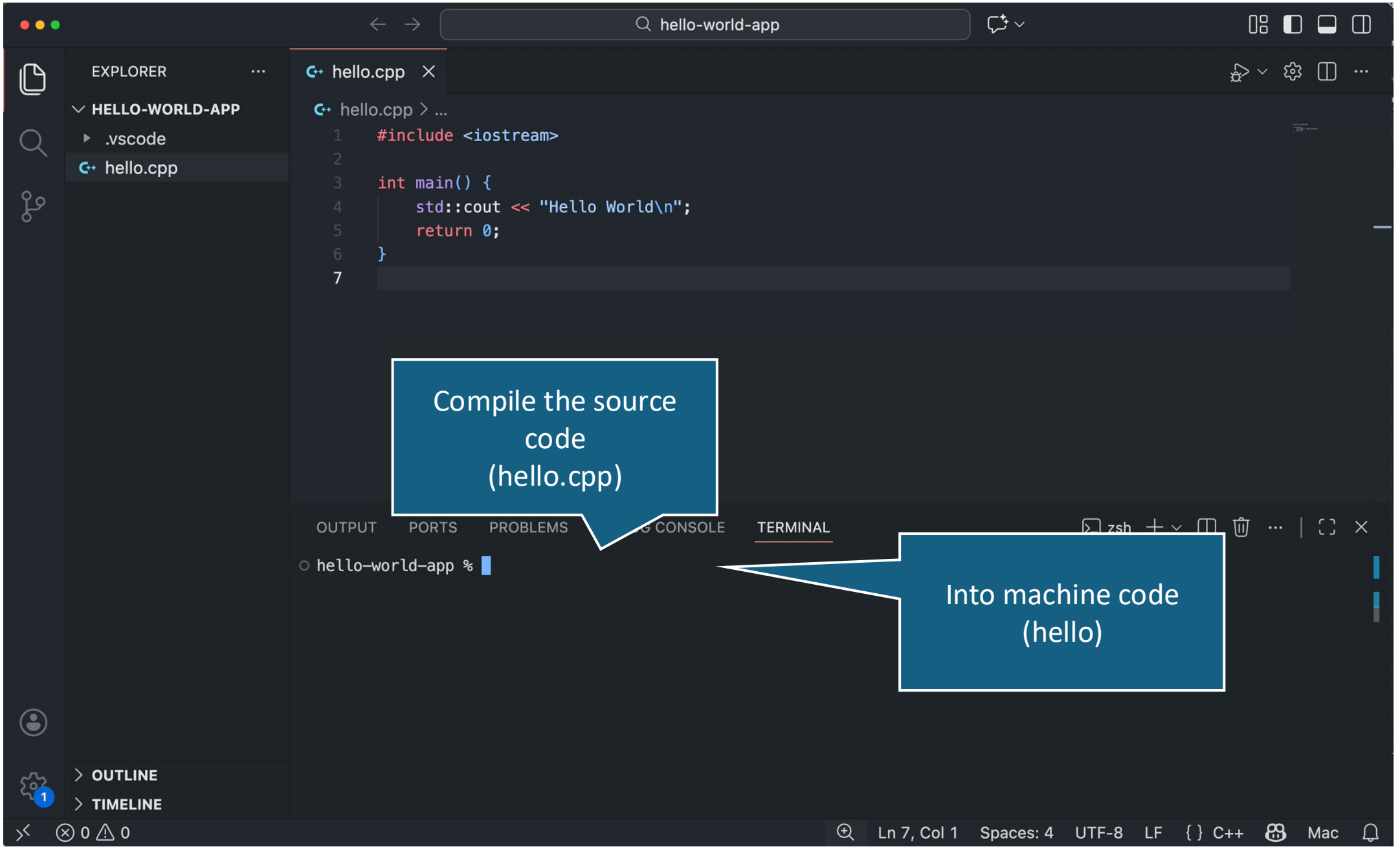
hello.cpp

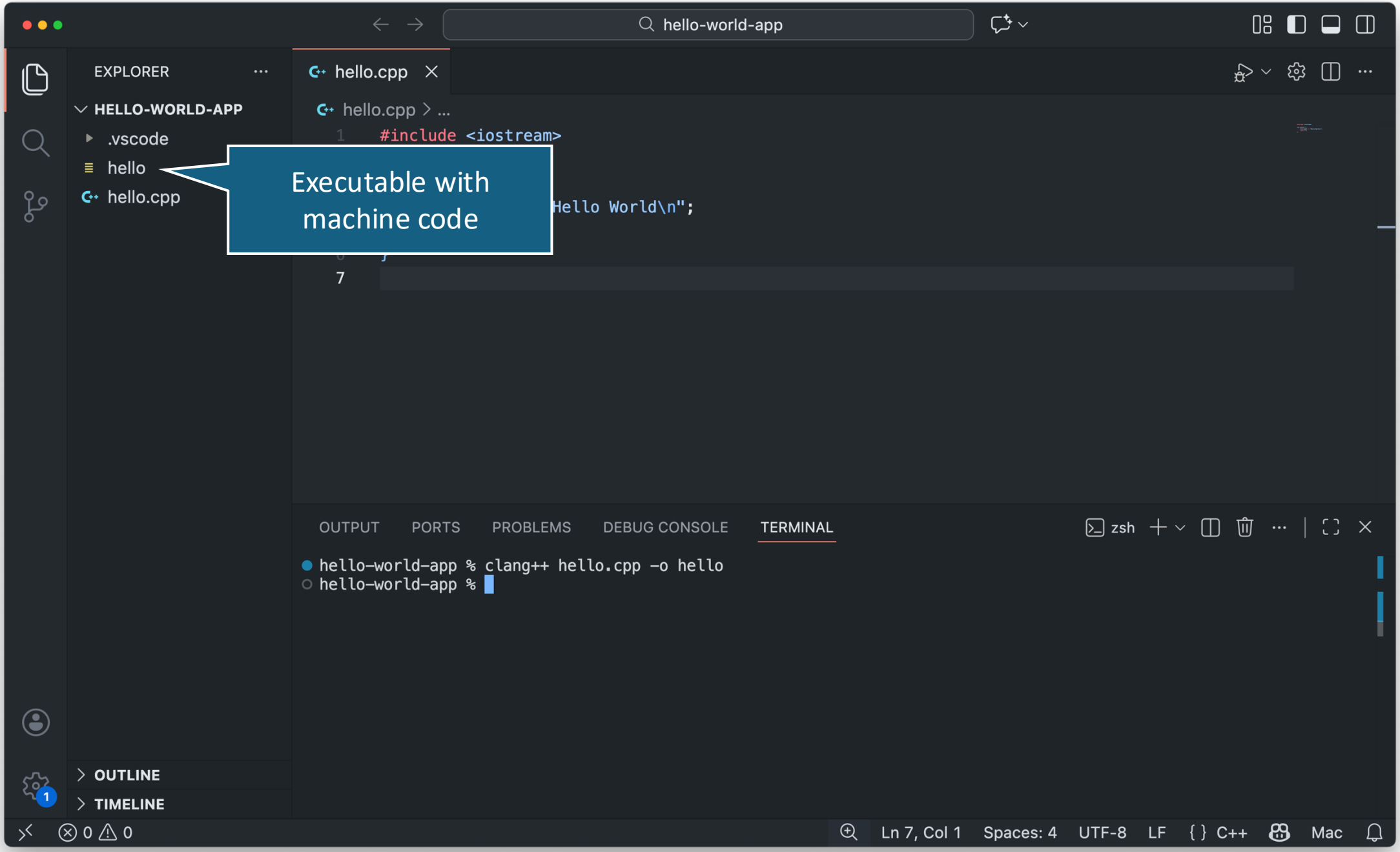
```cpp
#include <iostream>

int main() {
    std::cout << "Hello World\n";
    return 0;
}
```

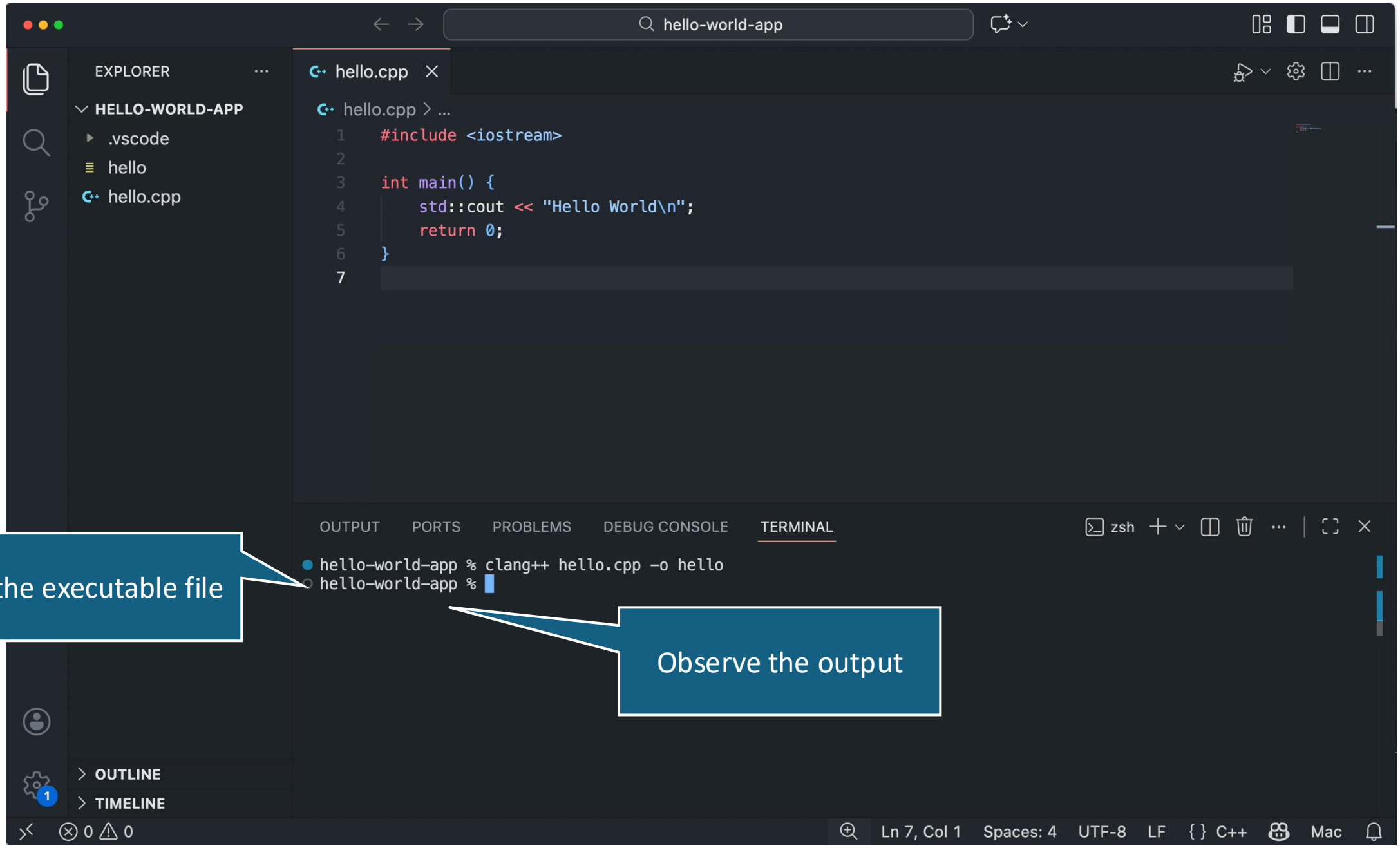
Compile



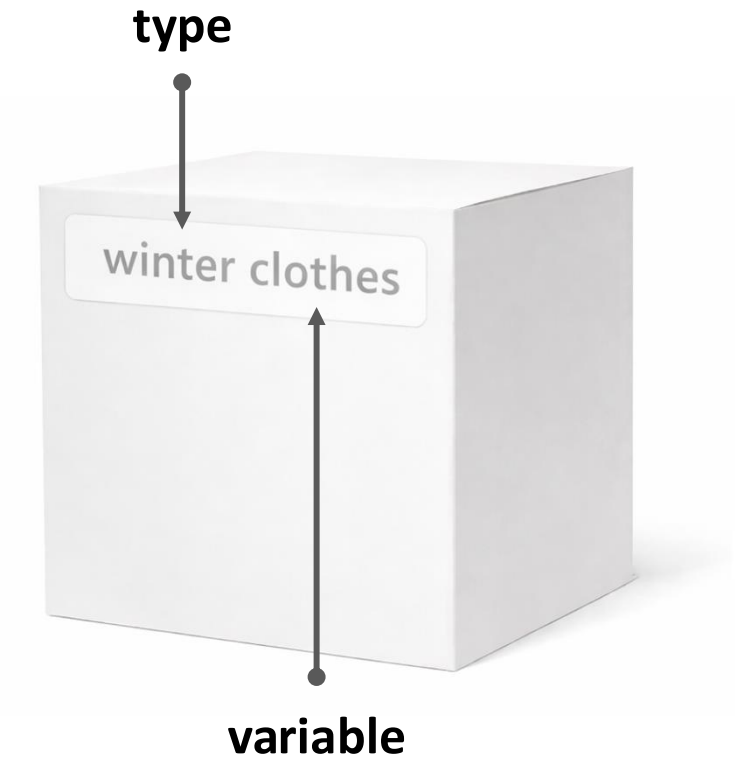

hello-world-app
EXPLORER
HELL...
.vscode
hello.cpp
My file
hello.cpp
hello.cpp > ...
#include <iostream>
int main() {
std::cout << "Hello World\n";
return 0;
}
My code
OUTPUT
PORTS
PROBLEMS
DEBUG CONSOLE
TERMINAL
zsh
hello-world-app %
OUTLINE
TIMELINE
Ln 7, Col 1
Spaces: 4
UTF-8
LF
C++
Mac


hello-world-app
EXPLORER
HELLO-WORLD-APP
.vscode
hello.cpp
hello.cpp
hello.cpp > ...
#include <iostream>
int main() {
std::cout << "Hello World\n";
return 0;
}
Compile the source code (hello.cpp)
OUTPUT
PORTS
PROBLEMS
CONSOLE
TERMINAL
zsh
hello-world-app %
Into machine code (hello)
OUTLINE
TIMELINE
Ln 7, Col 1
Spaces: 4
UTF-8
LF
C++
Mac


hello-world-app
EXPLORER
HELLO-WORLD-APP
.vscode
hello
hello.cpp
Executable with machine code
hello.cpp
hello.cpp > ...
1 #include <iostream>
Hello World\n";
7
OUTLINE
TIMELINE
OUTPUT
PORTS
PROBLEMS
DEBUG CONSOLE
TERMINAL
zsh
hello-world-app % clang++ hello.cpp -o hello
hello-world-app %
0 0
Ln 7, Col 1
Spaces: 4
UTF-8
LF
{} C++
Mac


hello-world-app
EXPLORER
HELLO-WORLD-APP
.vscode
hello
hello.cpp
hello.cpp
hello.cpp > ...
#include <iostream>
int main() {
std::cout << "Hello World\n";
return 0;
}
OUTPUT
PORTS
PROBLEMS
DEBUG CONSOLE
TERMINAL
zsh
hello-world-app % clang++ hello.cpp -o hello
hello-world-app %
Run the executable file
Observe the output
OUTLINE
TIMELINE
Ln 7, Col 1
Spaces: 4
UTF-8
LF
{ } C++
Mac

Storing data for analysis



- We store data into variables and assign them a data type.

```
int age = 22;
string name = “Fatima”;
```

Integer

Text

Data take memory space

Why assign a data type?

Computers need to know how much "space" (memory) to reserve.

News video — 5.000.000 bytes (5 MB)

Text: Stelios — 32 bytes

Number: 5.000.000 — 4 bytes

pi (3.14…) ~20 digits — 16 bytes

Exploring the fake news dataset

* Contains letters + numbers

string

* Decimal points (floats) numbers

double

id	title	read_length_min	ai_gen_score	flagged_fake
101	Flood hits coastal city overnight	2.5	0.76	true

int

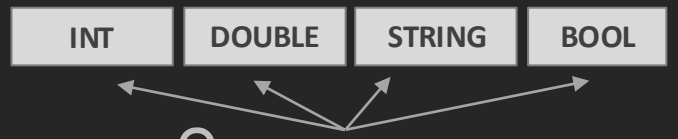
* Integer numbers

double

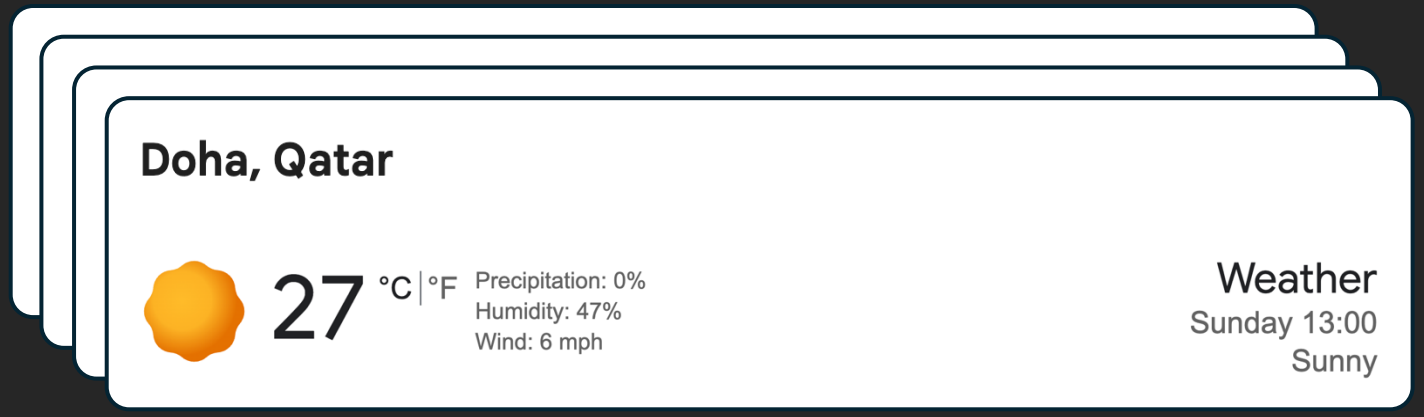
* Decimal points (floats) numbers

bool

* True or False data



Group quiz: What are the data labels and types?



Help me find a few!

city	country	temperature_c	percipitation	humidity	wind_mph	date_time	weather	raining
Doha	Qatar	27	0.00	0.47	6	Sunday, 13:00	Sunny	false
STRING	STRING	INT	DOUBLE	DOUBLE	INT	STRING	STRING	BOOL

What we learned by now

- What is C++?
  - Programming language
- What is a program?
  - A set of steps to solve a problem
- What is a compiler?
  - A program that translates source to machine code
- What are the basics data types?
  - int (whole), double (decimal), string (text), bool (true/false)

Break!

C++ Applied global challenges

PART 3

Fake News Analyzer ⟶ Building a simple analytics pipeline

- Programs become powerful when they react to **user input**.
  - Allows for dynamic data collection in the field or from laboratory devices.

Programming is about INPUT / OUTPUT

```
#include <iostream>

int main() {

    double ai_score;

    td::cout << "Enter AI-generation probability (0–1): ";

    td::cin >> ai_score;

    td::cout << "The article's AI probability is: " << ai_score;

    return 0;

}
```

Start

Declare variable

Output

Input

Store in variable

Output

End

Quiz (Individual)

Fill up the spaces

std::cin	double	0
std::cout	main	int

```
#include <iostream>

int _____() {
    _____ source_trust;
    _____ fact_score;
    _____ << "Enter source trust score (0–100): ";
    _____ >> source_trust;
    _____ << "Enter fact-check score (0–100): ";
    _____ >> fact_score;
    _____ credibility = (source_trust * 0.6) + (fact_score * 0.4);
    _____ << "Credibility score: " << credibility;
    return _____;
}
```

main
double
double
std::cout
std::cin
std::cout
std::cin
double
std::cout
0

What does this program do?

```
#include <iostream>

using namespace std;

int main() {
    double a, b;
    cout << "Enter two numbers: ";
    cin >> a >> b;
    cout << "Result: " << a / b;
    return 0;
}
```

Asks the user to enter two numbers and display their division on the screen.

- What is the problem with this code? Turn to the person next to you and discuss
- What if the inputs are `5` and `0` ?
- **Denominator (`b`) cannot be zero!**

Decision Logic

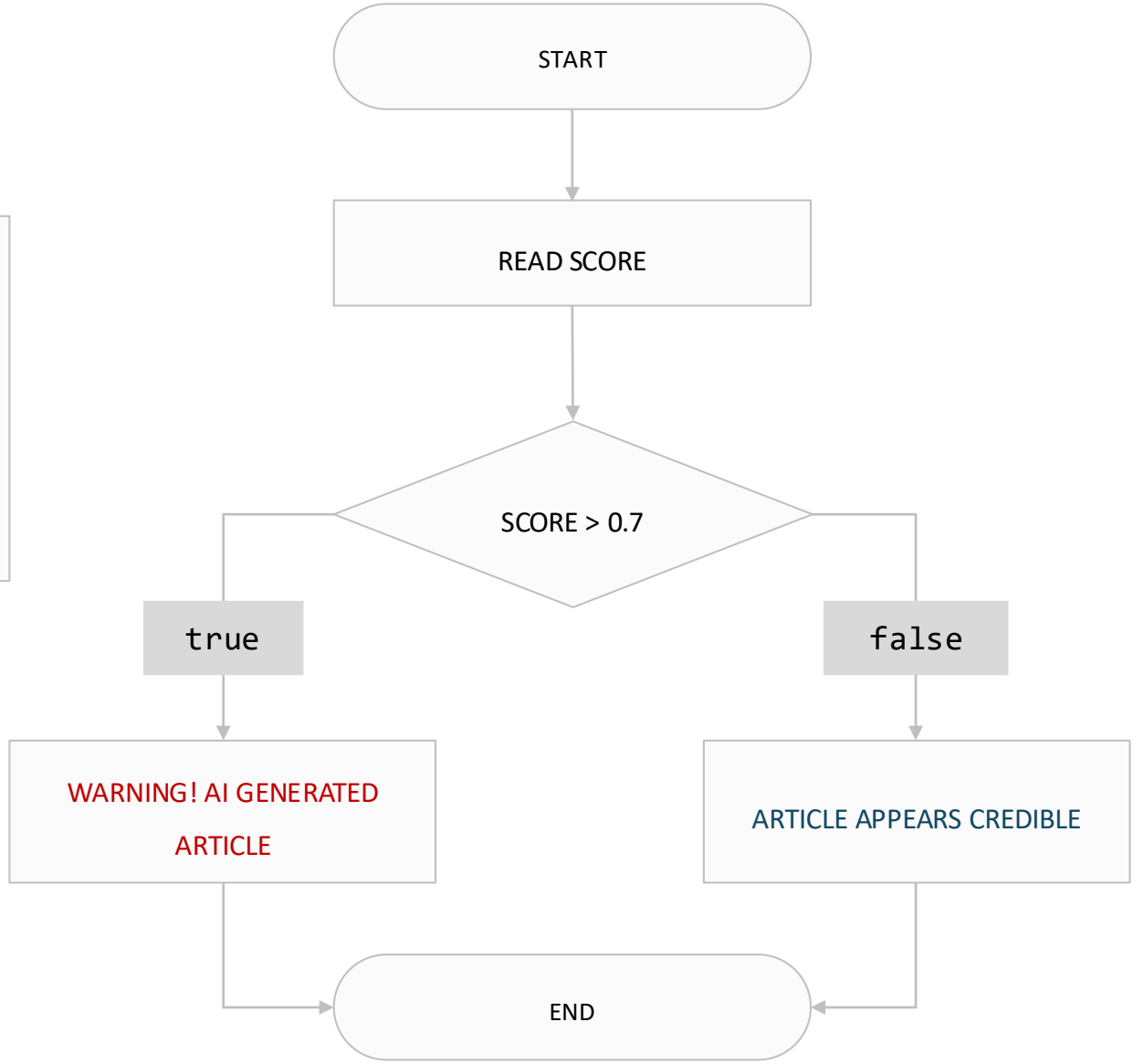
- We use logic to define "rules" for our systems
- Allows the program to choose between different actions
- The foundation of automated decision support

What if the AI score is very high?

Should we warn the user?

Decision Logic

If an article's "fake score" is **greater than 70%**, inform the user accordingly.



Example script

if runs code when a condition is **true**.

else runs code when the condition is **false**.

```
double ai_score;

cout << "Enter AI probability score (0–1): ";

cin >> ai_score;

if (ai_score > 0.7)

{ std::cout << "Warning: This article is likely AI-generated or fake."; }

else

{ std::cout << "This article appears credible."; }
```

C++ Comparison Operators

Comparison operators are used to compare two values.

They always return **true** or **false**.

Operator	Meaning	Example	Result
==	equal to	5 == 5	true
!=	not equal to	5 != 3	true
>	greater than	7 > 3	true
<	less than	22 < 10	false
>=	greater than or equal to	5 >= 5	true
<=	less than or equal to	4 <= 3	false

Let's fix the division

```
#include <iostream>

using namespace std;


int main() {

    double a, b;

    cout << "Enter two numbers: ";

    cin >> a >> b;

    cout << "Result: " << a / b;

    return 0;

}
```

```
#include <iostream>

using namespace std;


int main() {

    double a, b;

    cout << "Enter two numbers: ";

    cin >> a >> b;

    if (b == 0)

   { cout << "Error: cannot divide by zero.” }

    else

   { cout << "Result: " << a / b }

    return 0;

}
```

You might have more than two decisions

- Given an AI probability score, classify an article according it's ai_score.

LOW	MEDIUM	HIGH
0-39%	40%-79%	80%-100%

* Assume numbers are always between 0 and 1.

```
if (ai_score >= 0.8)
{ cout << "High risk ↑"; }
else if (ai_score >= 0.4)
{ cout << "Medium risk →"; }
else
{ cout << "Low risk ↓"; }
```

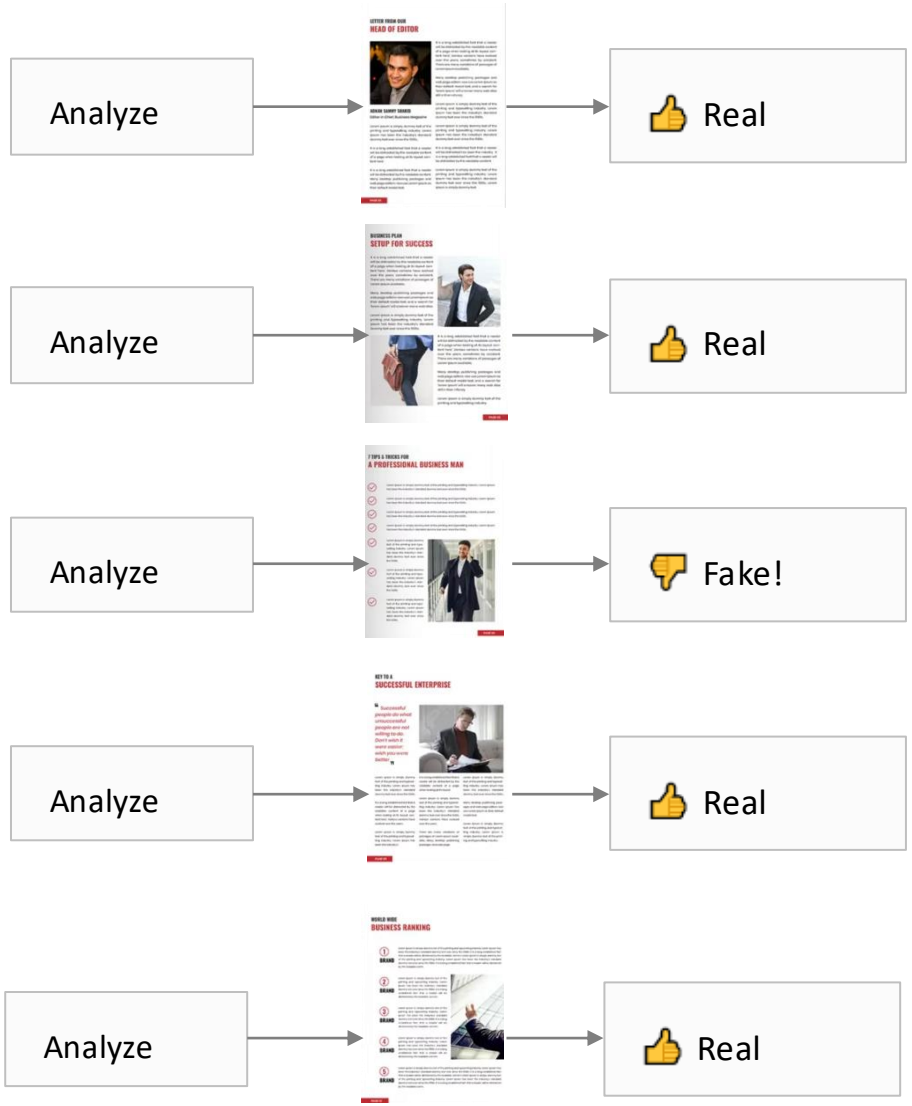
We rarely analyze just one news article

PROBLEM

- What if we have 100s of articles?

SOLUTION

- Repeatedly examine multiple articles and flag those likely to be fake.



Building loops

- A loop repeats the same instructions multiple times.

```
for (int i = 1; i <= 5; i++) {
    std::cout << i;
}
```

for	→ repeat code
i = 1	→ start counting at 1
i <= 5	→ stop after 5 times
i++	→ increase by 1 each time
cout	→ prints the message

* Loop runs **the same code five times**.

Scan 5 articles and check their AI-probability scores

- Use a **loop** to repeat instructions

More to come soon!

```
for (int id = 1; id <= 5; id++) {
   ...
   if (ai_score > 0.7)
   { ..."is likely fake."... }

   else
   { ..."seems credible." ... }
}
```

Interactive exercises

PART 4

Exercise1

Pair programming exercise

In this activity, you will build a simple C++ program that **simulates how a government agency monitors cyber threat levels and decides what action to take**.

1. Enter a cyber threat score (0–100).
2. Check 5 scores in total.
3. Use `if`, `else if`, `else` to label each as **low**, **medium**, or **high**.
4. Print what the agency should do for each score.
5. Try different values and see how decisions change.

> 👥 Work in pairs to write the code, test different inputs, and discuss how changing the thresholds can affect real-world decisions.

Exercise1 — Edited

```cpp
#include <iostream>
using namespace std;

int main() {
    int score;

    // check 5 scores
    for (int i = 1; i <= 5; i++) {
        cout << "Enter cyber threat score (0-100): ";
        cin >> score;

        if (score >= 70) {
            cout << "High threat → Activate emergency protocol\n";
        }
        else if (score >= 40) {
            cout << "Medium threat → Increase monitoring\n";
        }
        else {
            cout << "Low threat → Normal operations\n";
        }

        cout << endl;
    }

    cout << "All scores checked." << endl;

    return 0;
}
```

Reflection

PART 5

Discuss examples of interdisciplinary impact

- Computer science + policy

  → misinformation regulation: fake news and digital policies

- Computer science + climate science

  → environmental modelling: predict floods, heatwaves, or rising sea levels

- Computer science + healthcare

  → medical diagnostics: analyse medical images or patient data

- Computer science + business

  → risk analytics: detect fraud and manage investment risk

Key Takeaways

- Programming = problem solving
- Input → decision → output
- Loops + conditionals for real data
- Code can address real-world issues
- Example: detecting misinformation

Summarizing Teaching Approach

- Start with real-world problem
- Build concepts step by step
- Ask questions and predict outcomes
- Short interactive activity
- Connect to multiple disciplines

Questions?

“Everybody should learn how to program a computer, because it teaches you how to think”

Steve Jobs