
OPENTECHZ
A dream to realize your dream
C++
By Parthasarathi Swain

Session Objectives

- Understand the mindset required to learn programming
- Learn to compare programming with English
- Appreciate the value of practice and patience

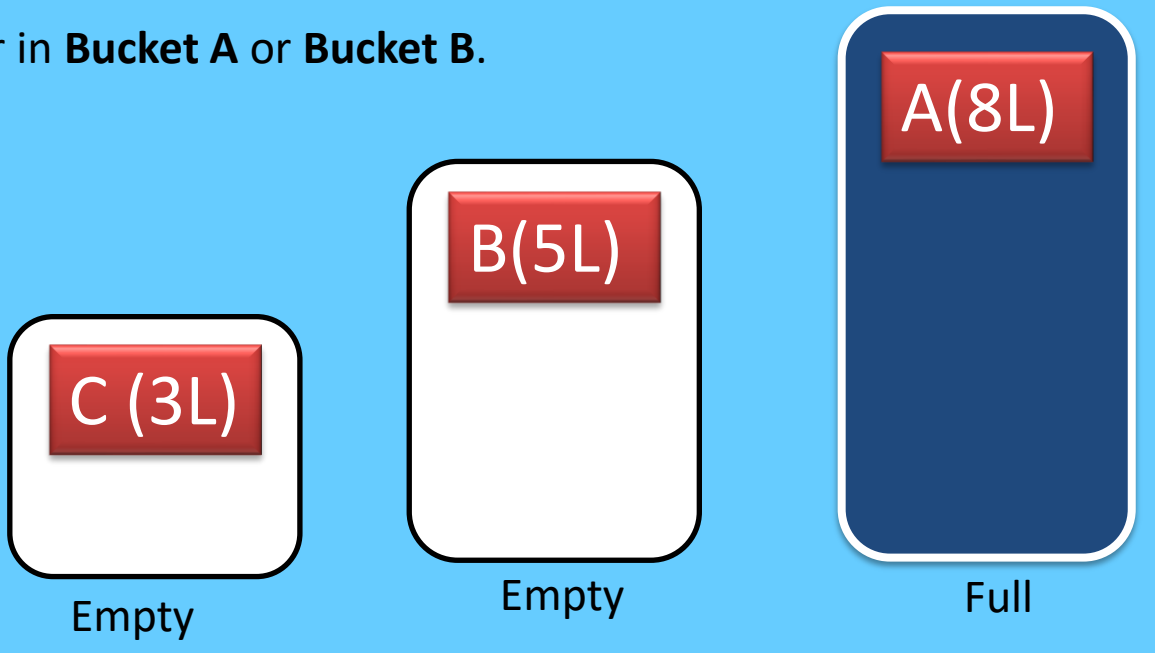
IQ Test

Problem Statement:

- You have three buckets:
- **Bucket A**: 8 liters (full)
- **Bucket B**: 5 liters (empty)
- **Bucket C**: 3 liters (empty)

Goal:

Measure **exactly 4 liters** of water in **Bucket A** or **Bucket B**.



Real-Life Scenario – Learning Cycle

Example: How a child learns a language:

- Observes letters and sounds (A, B, C)
- Forms words (Cat, Dog)
- Makes sentences (I love coding)
- Learns grammar (Tenses, Punctuation)
- Writes essays and stories

Same in Programming:

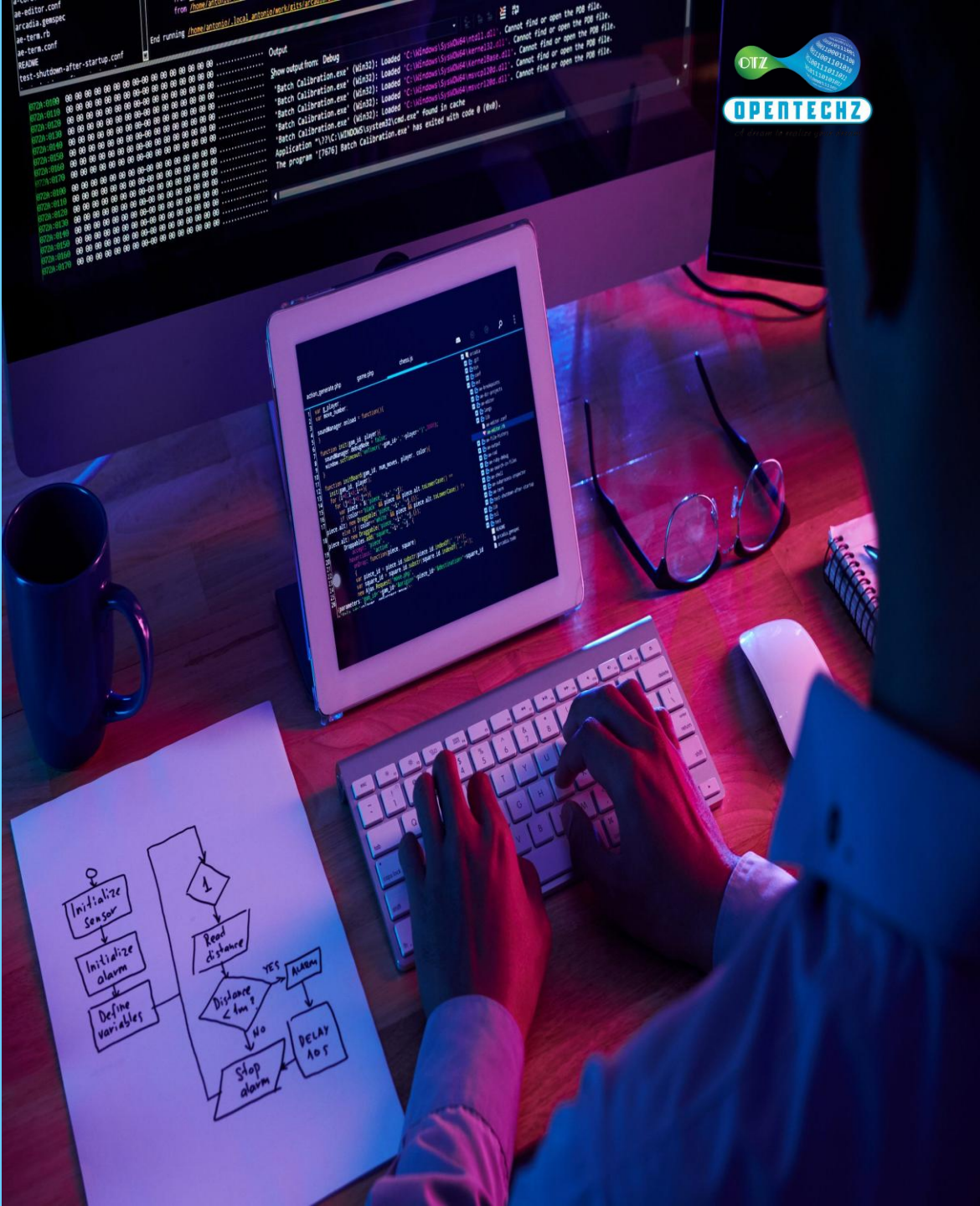
Characters → Tokens → Statements → Syntax → Full Program

Programming vs English Learning

English Learning	Programming Learning
Alphabets (A-Z)	Characters (A-Z, 0-9, symbols)
Words	Keywords, Identifiers
Sentences	Code Statements
Grammar	Syntax Rules
Composition	Complete Program

How to Learn Programming Effectively

- Start with understanding logic, not just syntax
- Write code yourself, don't just copy-paste
- Break complex tasks into simpler parts
- Learn from each error; debug patiently



Importance of Practice

Real Example:

- Like learning to type fast – more practice = better speed
- In C++, more problems = better logic

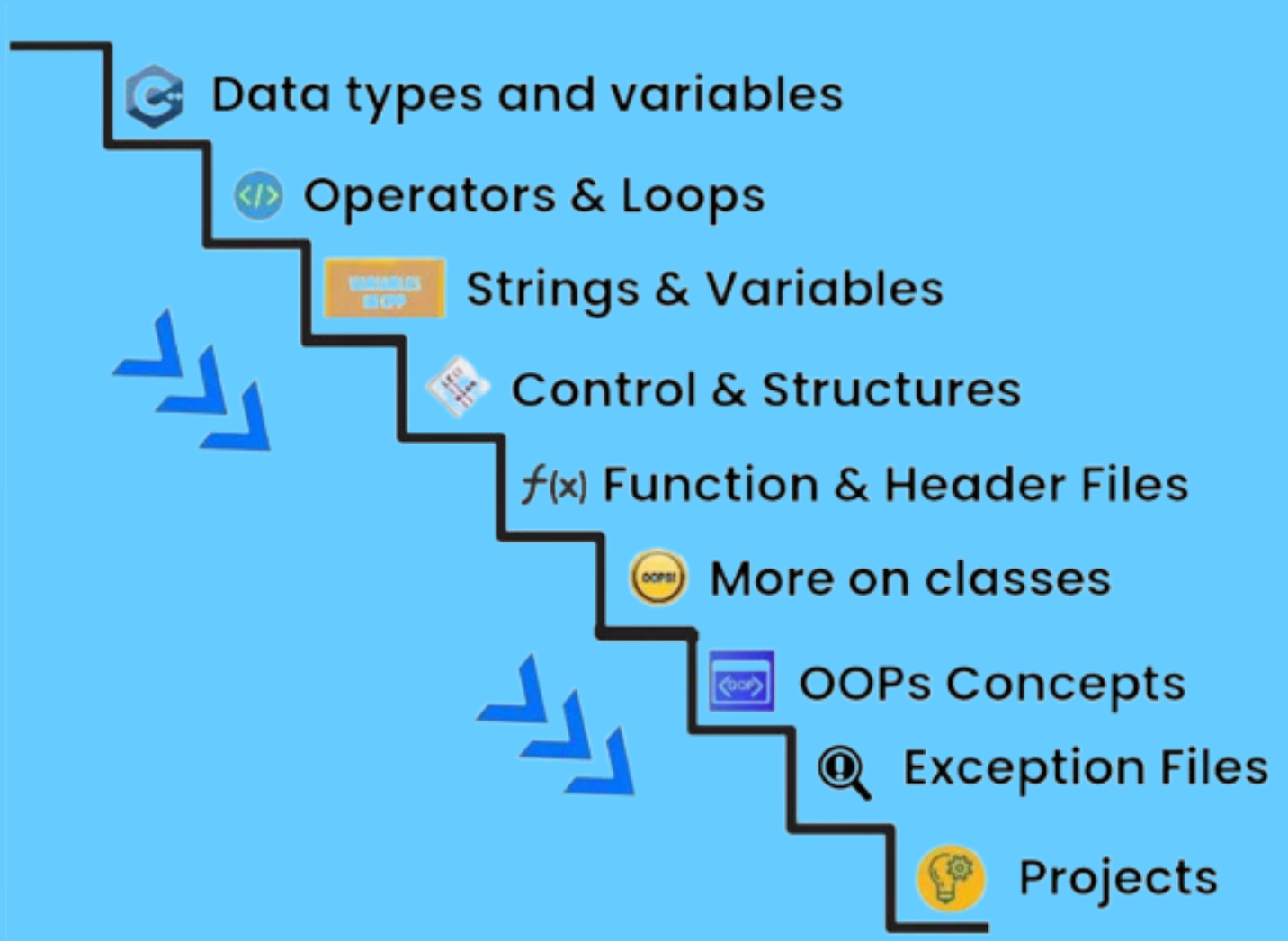
Tips:

- Solve beginner programs (sum, max, even/odd)
- Use platforms like HackerRank, GeeksforGeeks
- Code at least 30 minutes every day

Be Patient – It Takes Time

- ✓ Programming is a journey, not a sprint
- ✓ Errors are your best teachers
- ✓ Success takes time and trials

Roadmap to Learn C++



What is C++?

- C++ is a **high-level programming language**.
- Developed by **Bjarne Stroustrup** in 1983.
- It is **fast**, **powerful**, and used in:
  - ✓ Game development
  - ✓ Operating systems (like **Windows**, **Linux**)
  - ✓ Banking and financial systems
  - ✓ Robotics and embedded systems
  - ✓ Real-time and performance-critical applications

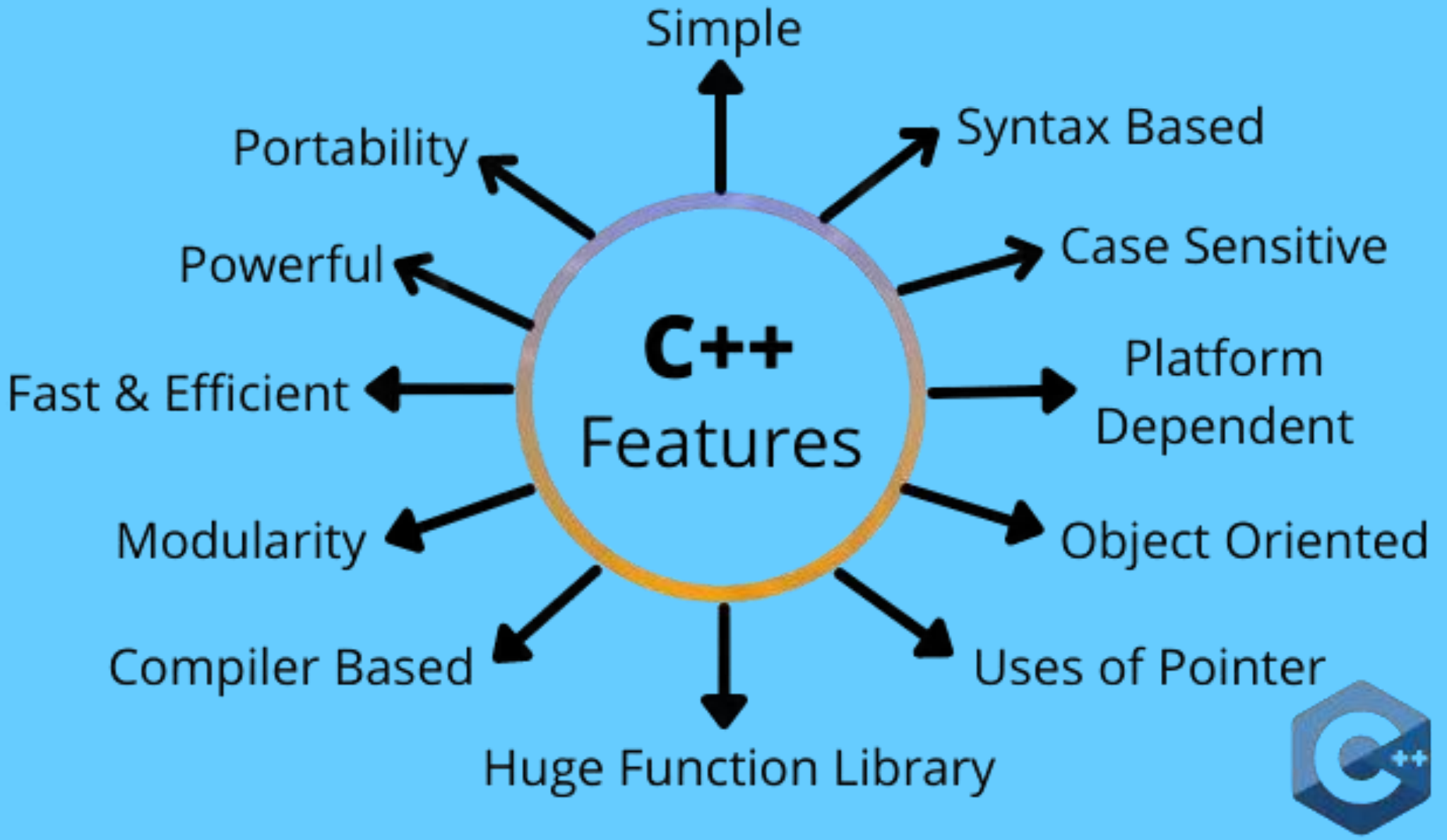
Why Choose C++?

- ❖ C++ is fast, efficient, and close to hardware
- ❖ Great for understanding how computers work
- ❖ Builds strong base for other languages (Java, Python)

Real Use Cases:

- Game Engines
- Embedded Systems (microcontrollers)
- Operating Systems (Windows parts)

Key Features of C++

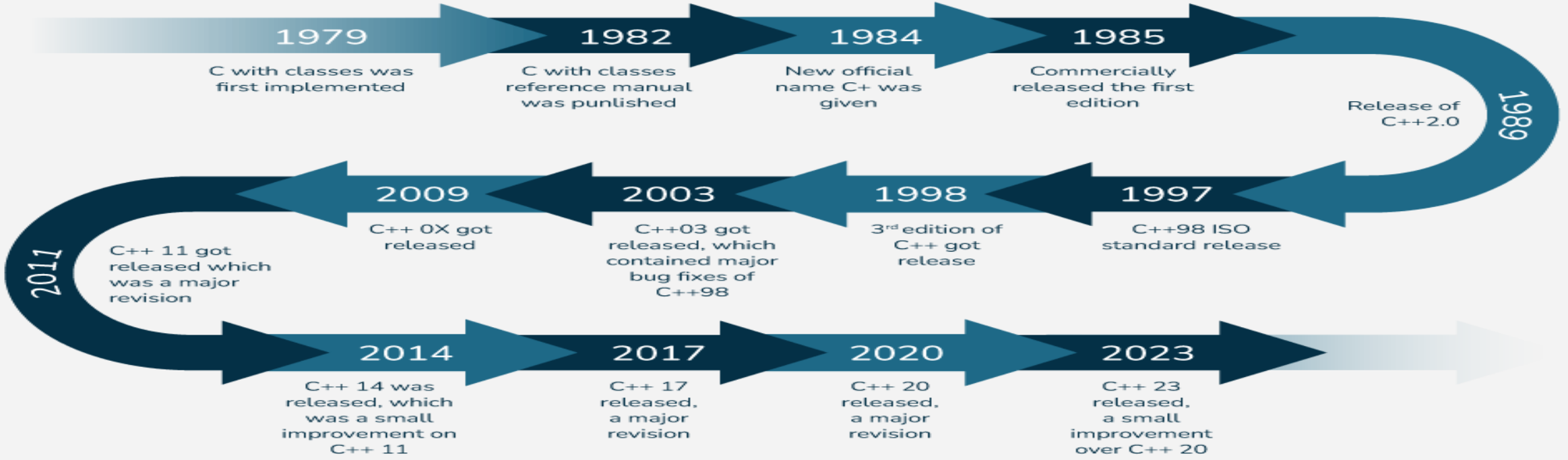


C++ vs Other Languages

Language	Speed	Syntax	Usage
C++	Very Fast	Medium	Games, OS, Embedded
Python	Slower	Easy	AI, ML, Data Science
Java	Fast	Moderate	Web, Enterprise Apps

History of C++

- **1979** – Bjarne Stroustrup began developing C++ at **Bell Labs** (USA).
- **1983** – The language was first named **“C with Classes”**, later renamed **C++**.
- **C++** was built as an extension of **C**, adding **object-oriented features**.



Installation of C++

Step 1 – Download Dev C++

- ✓ Visit: https://sourceforge.net/projects/orwelldevcpp/
- ✓ Click on the **Download** button
- ✓ File size: ~50MB

Step 2 – Run Installer

- ✓ Locate the downloaded .exe file
- ✓ Double-click to start installation
- ✓ Choose preferred **language**
- ✓ Click **Next**

Step 3 – Accept License

- ✓ Read the License Agreement
- ✓ Click **I Agree** to continue

Installation of C++

Step 4 – Choose Installation Folder

- ✓ Default path is: C:\Program Files (x86)\Dev-Cpp
- ✓ You can change the folder if needed
- ✓ Click **Install**

Step 5 – Finish Installation

- ✓ Wait for installation to complete
- ✓ Click **Finish**
- ✓ Dev C++ will launch automatically

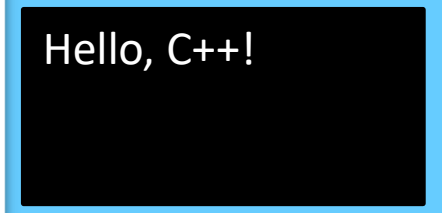
Verify Installation

Step 7 – Verify Installation

Go to **File > New > Source File**

Type and run this code:

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello, C++!";
    return 0;
}
```



Thank You