XX00FC41-3001

C++ PROGRAMMING

Lecture 01 -Wednesday 2022/08/31

TODAY

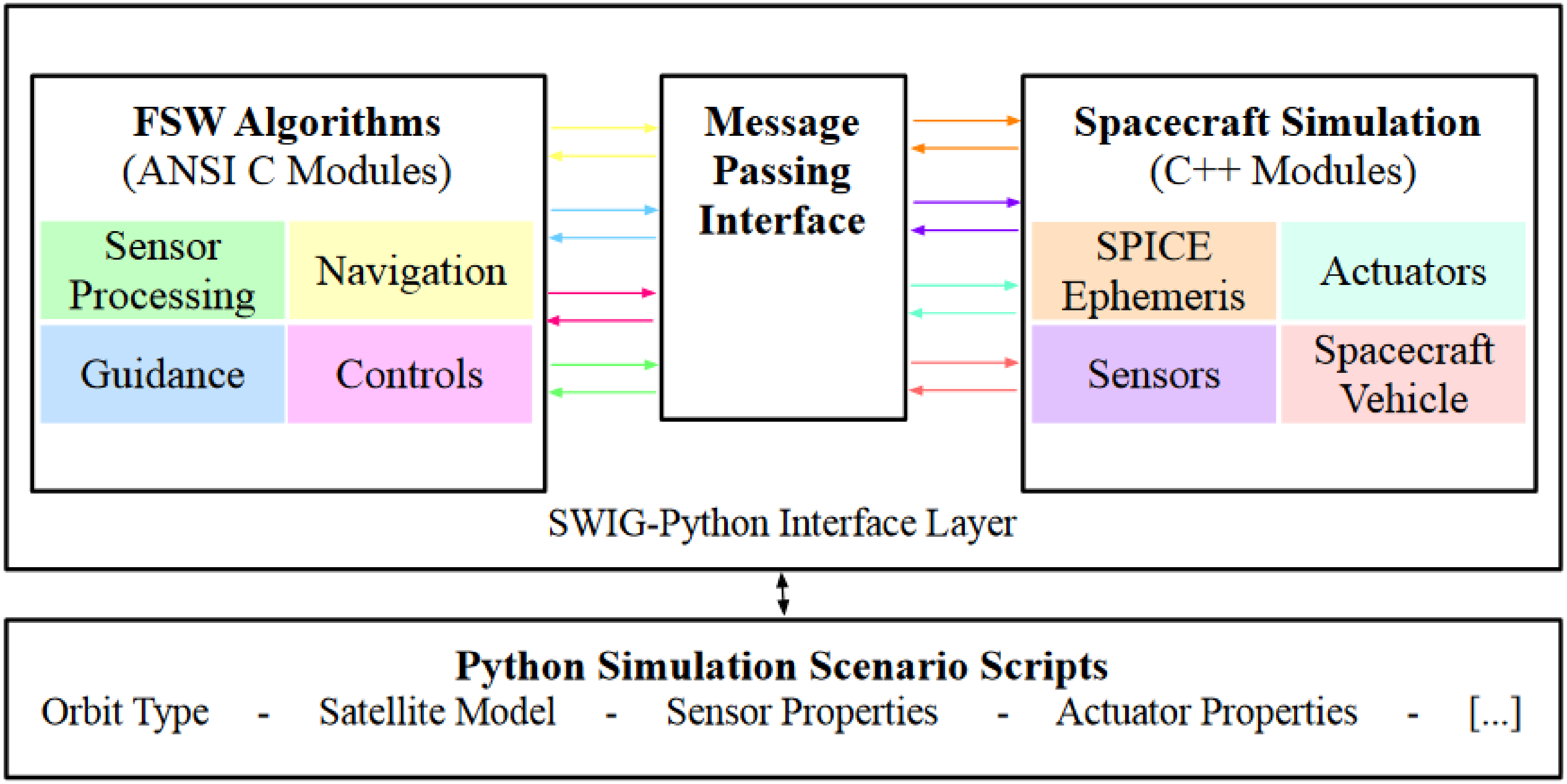
- Problem Solving
- Input / Output
- Objects, Types, and Values
- Operations and operators
- Assignment and initialization

USE CASES

- Toys
- Space
- Aeronautics
- Intravehicular Distributed Systems
- Gaming etc.

C++ W PYTHON

Modular Platform for Hardware-in-the-Loop Testing

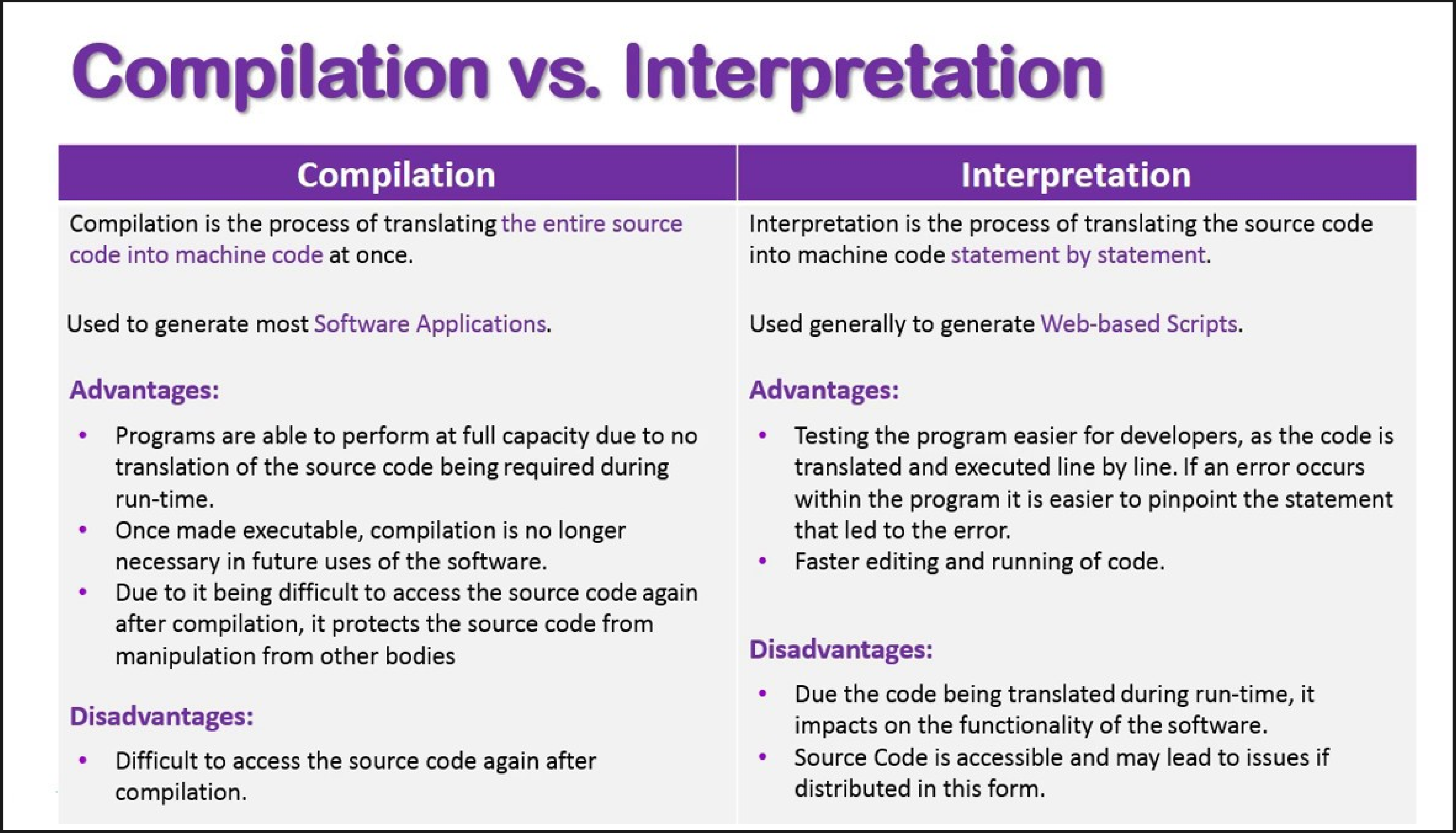

FSW Algorithms
(ANSI C Modules)
Sensor Processing
Navigation
Guidance
Controls
Message Passing Interface
Spacecraft Simulation
(C++ Modules)
SPICE Ephemeris
Actuators
Sensors
Spacecraft Vehicle
SWIG-Python Interface Layer
Python Simulation Scenario Scripts
Orbit Type - Satellite Model - Sensor Properties - Actuator Properties - [...]

HELLO WORLD!

```
#include ❰iostream❱

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

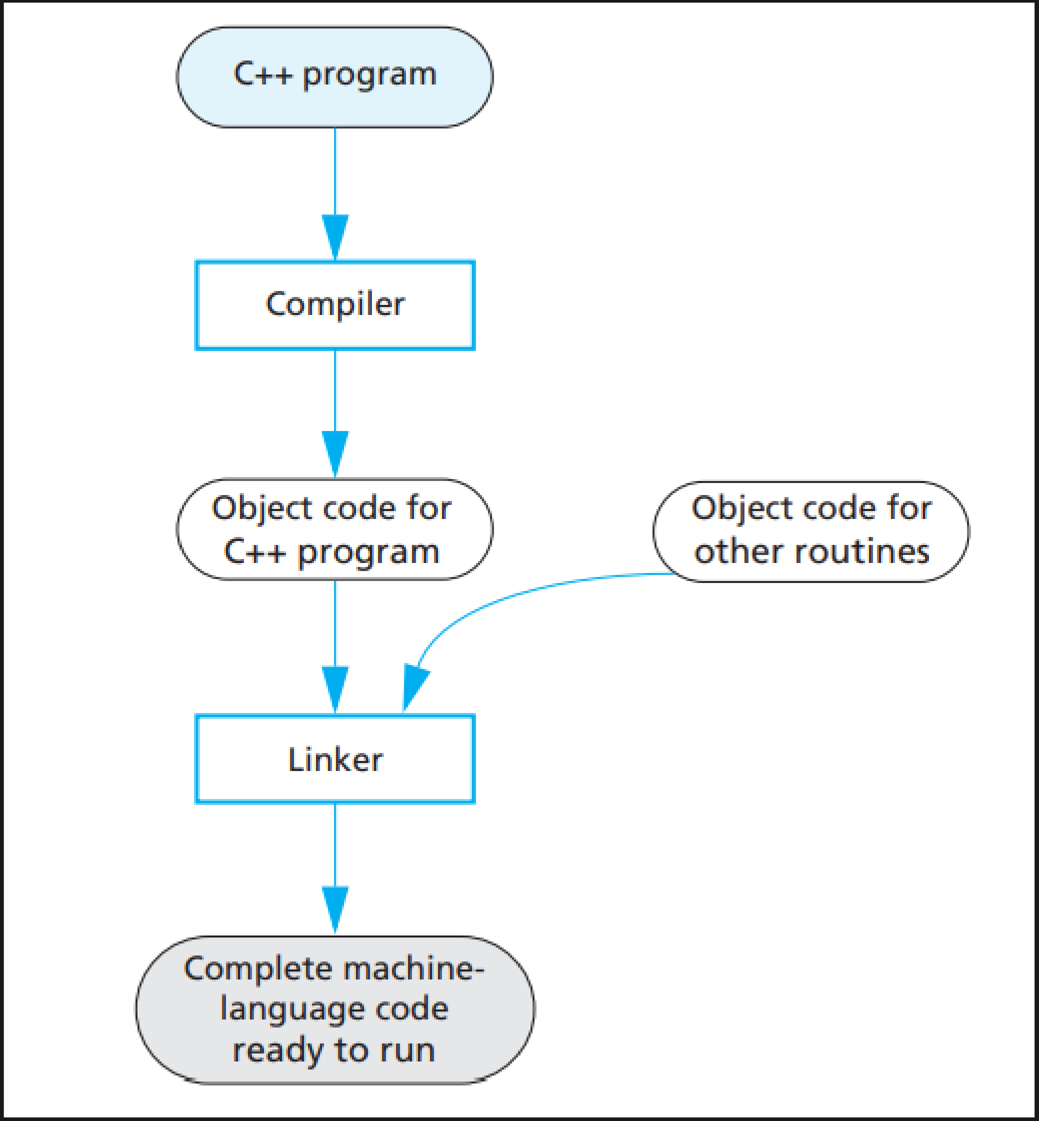
INTERPRETED VS COMPILED



Compilation vs. Interpretation

Compilation	Interpretation
Compilation is the process of translating the entire source code into machine code at once.	Interpretation is the process of translating the source code into machine code statement by statement.
Used to generate most Software Applications.	Used generally to generate Web-based Scripts.
Advantages: • Programs are able to perform at full capacity due to no translation of the source code being required during run-time. • Once made executable, compilation is no longer necessary in future uses of the software. • Due to it being difficult to access the source code again after compilation, it protects the source code from manipulation from other bodies	**Advantages:** • Testing the program easier for developers, as the code is translated and executed line by line. If an error occurs within the program it is easier to pinpoint the statement that led to the error. • Faster editing and running of code.
Disadvantages: • Difficult to access the source code again after compilation.	**Disadvantages:** • Due the code being translated during run-time, it impacts on the functionality of the software. • Source Code is accessible and may lead to issues if distributed in this form.

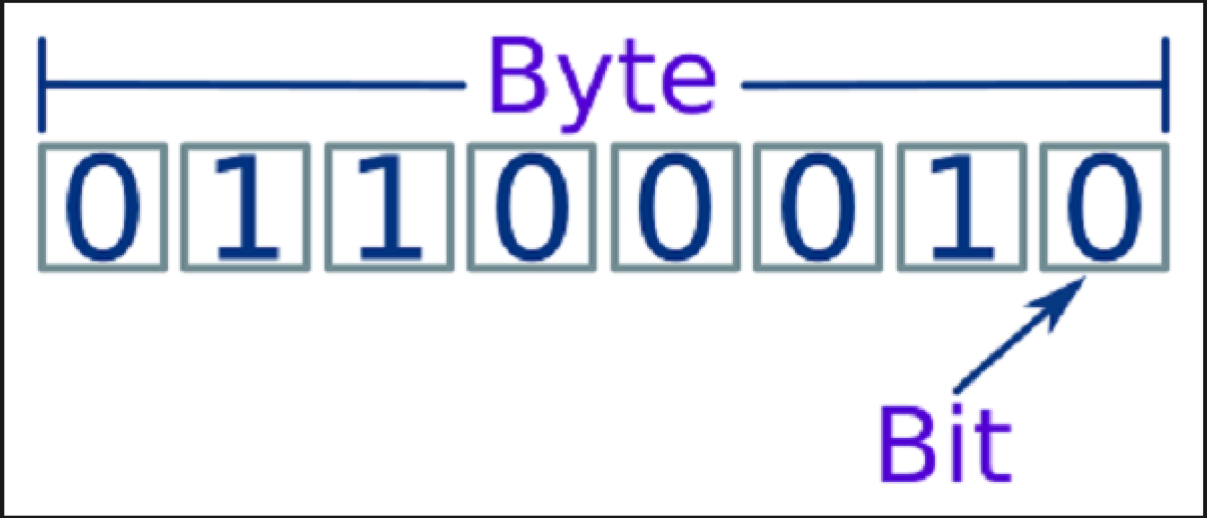
RUNNING C++ PROGRAM



C++ program
Compiler
Object code for C++ program
Object code for other routines
Linker
Complete machine-language code ready to run

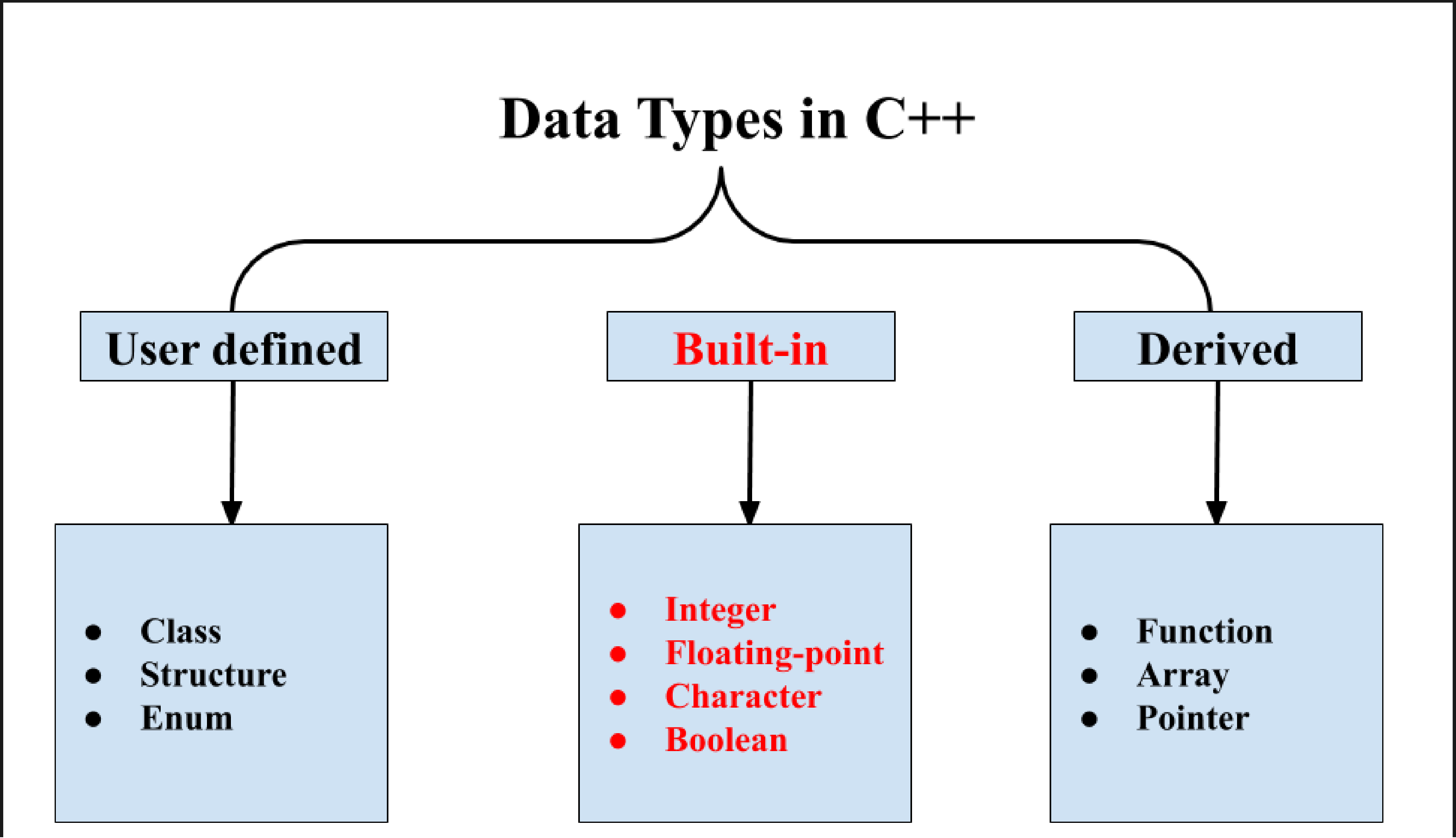
MACHINE CODE

BITS & BYTES



DATA TYPES

What about strings?



Data Types in C++

User defined

- Class
- Structure
- Enum

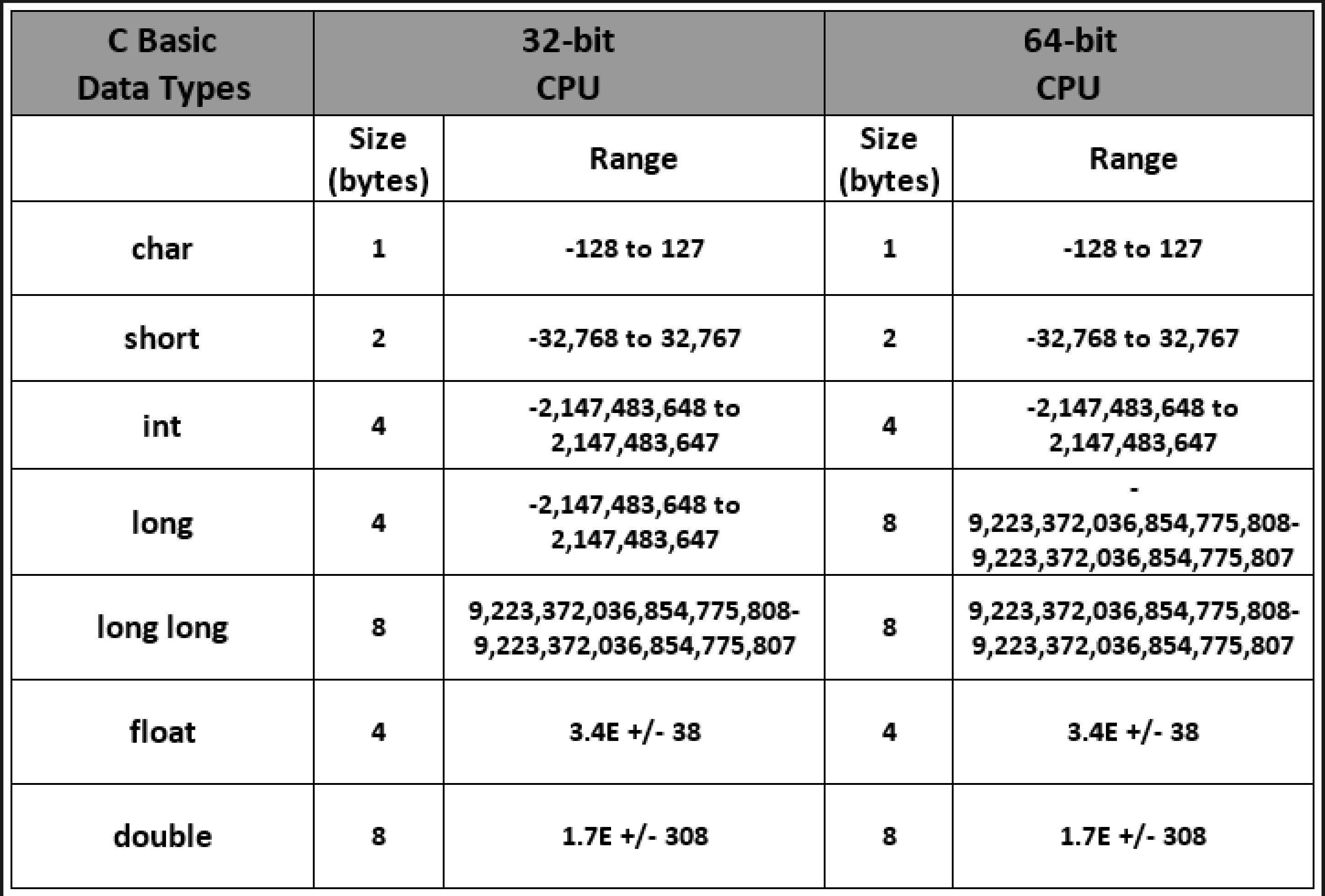
Built-in

- Integer
- Floating-point
- Character
- Boolean

Derived

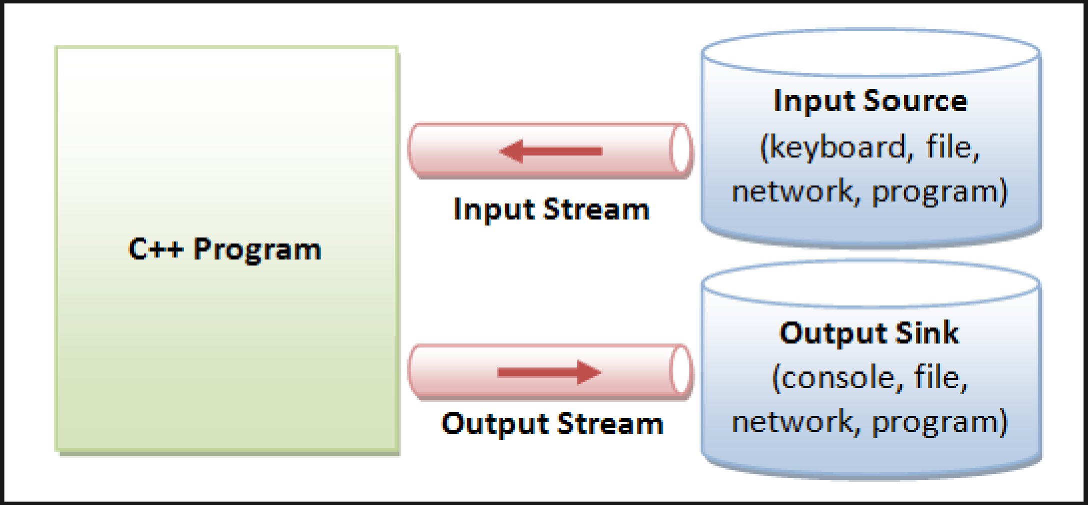
- Function
- Array
- Pointer

DATA TYPES SIZES

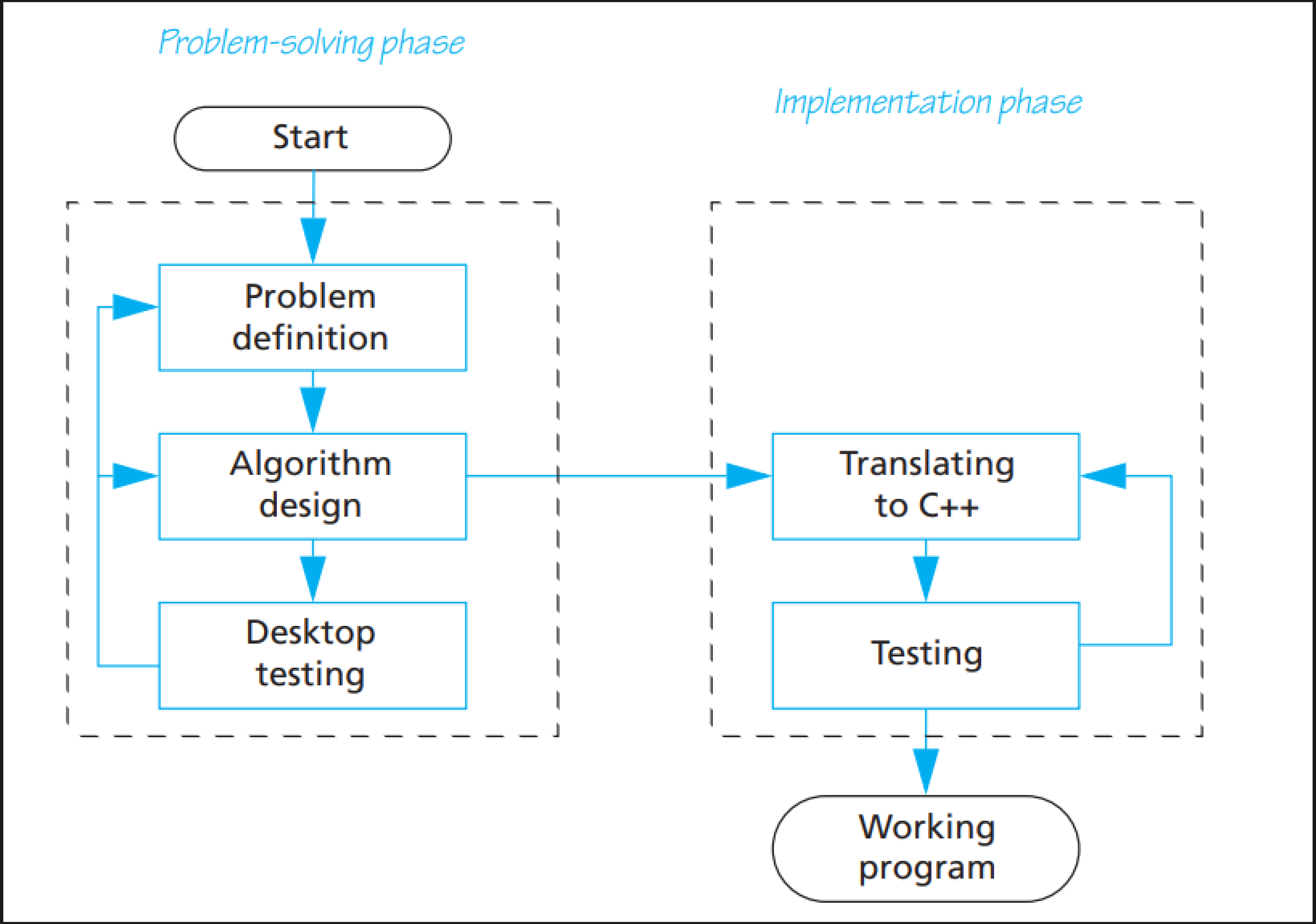


C Basic Data Types	32-bit CPU		64-bit CPU	
	Size (bytes)	Range	Size (bytes)	Range
char	1	-128 to 127	1	-128 to 127
short	2	-32,768 to 32,767	2	-32,768 to 32,767
int	4	-2,147,483,648 to 2,147,483,647	4	-2,147,483,648 to 2,147,483,647
long	4	-2,147,483,648 to 2,147,483,647	8	- 9,223,372,036,854,775,808- 9,223,372,036,854,775,807
long long	8	9,223,372,036,854,775,808- 9,223,372,036,854,775,807	8	9,223,372,036,854,775,808- 9,223,372,036,854,775,807
float	4	3.4E +/- 38	4	3.4E +/- 38
double	8	1.7E +/- 308	8	1.7E +/- 308

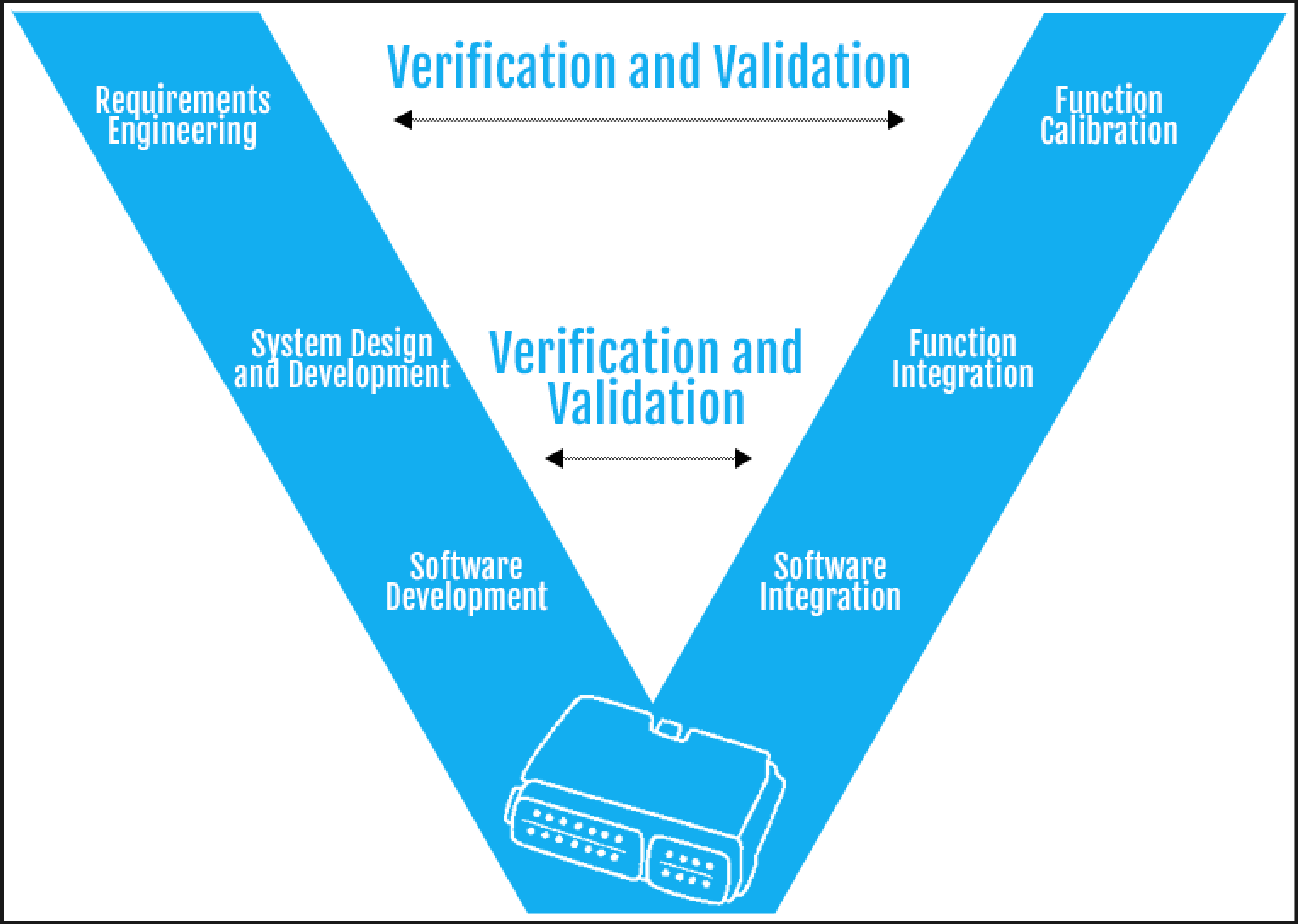
INPUT / OUTPUT



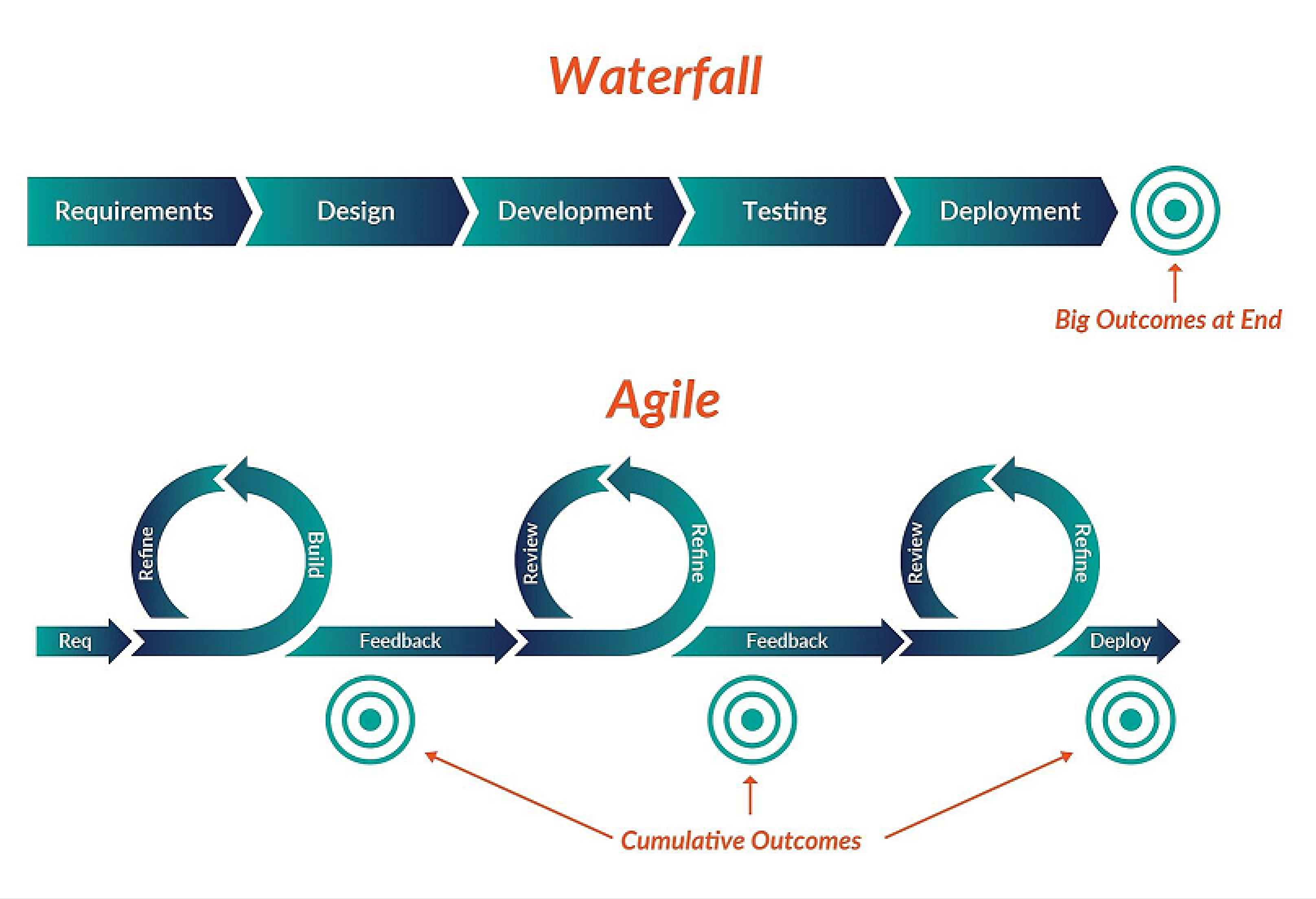
PROGRAM DESIGN PROCESS



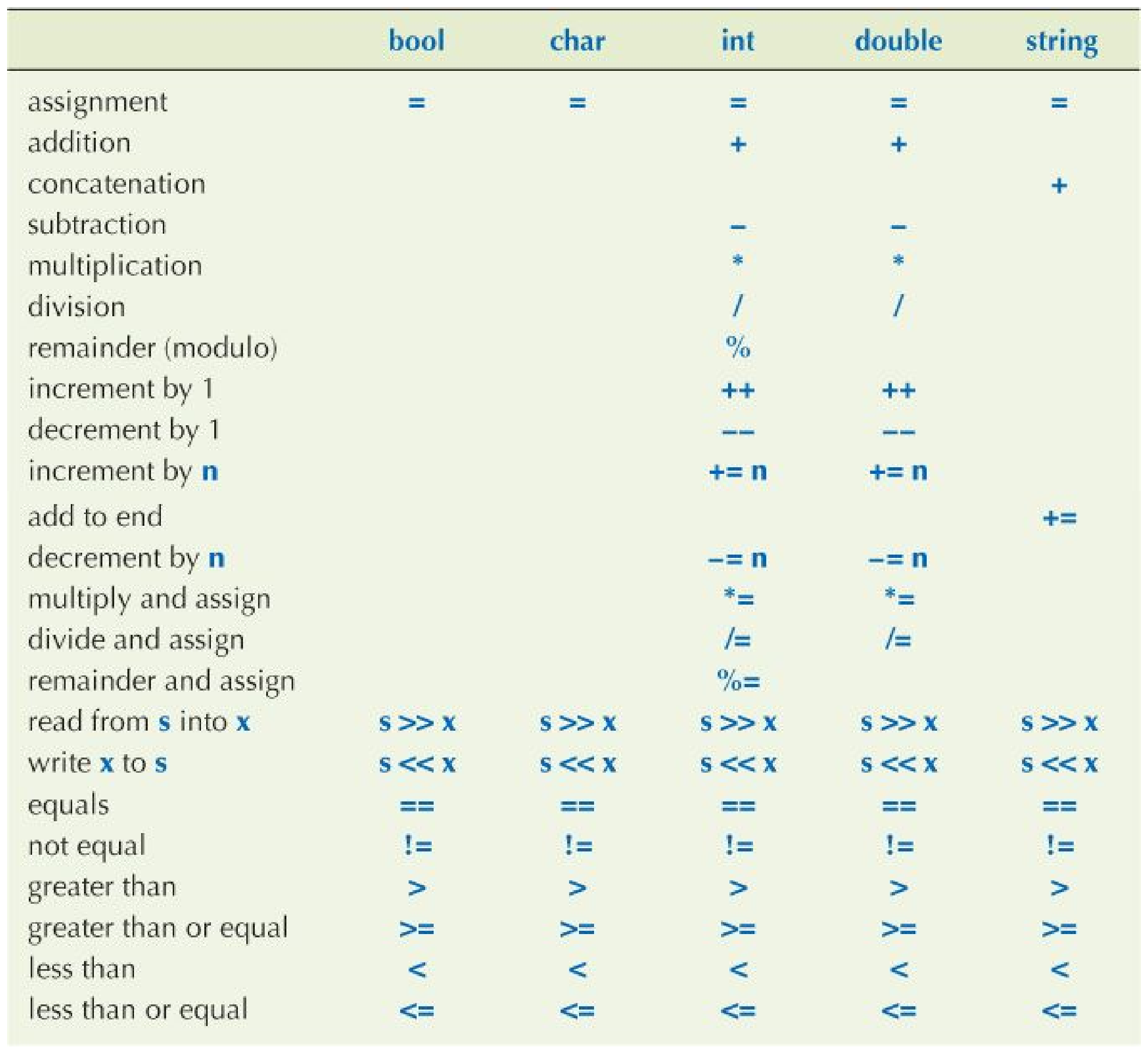
PROGRAM DESIGN PROCESS...



PROGRAM DESIGN PROCESS...



OBJECTS, TYPES, AND VALUES



	bool	char	int	double	string
assignment	=	=	=	=	=
addition			+	+	
concatenation					+
subtraction			–	–	
multiplication			*	*	
division			/	/	
remainder (modulo)			%		
increment by 1			++	++	
decrement by 1			––	––	
increment by **n**			+= n	+= n	
add to end					+=
decrement by **n**			–= n	–= n	
multiply and assign			*=	*=	
divide and assign			/=	/=	
remainder and assign			%=		
read from **s** into **x**	s >> x	s >> x	s >> x	s >> x	s >> x
write **x** to **s**	s << x	s << x	s << x	s << x	s << x
equals	==	==	==	==	==
not equal	!=	!=	!=	!=	!=
greater than	>	>	>	>	>
greater than or equal	>=	>=	>=	>=	>=
less than	<	<	<	<	<
less than or equal	<=	<=	<=	<=	<=

Thank You!