CSC10001 – Introduction to Programming

1st lecture: Introduction to Computers and Programming

fit hcmus
Instructor: TRAN Thi Thao Nhi

October 2, 2023

Information

Instructors:

- Tran Thi Thao Nhi
- Phan Thi Phuong Uyen
- Nguyen Van Quang Huy

Time and place: `13:30 - 17:10, I41`

Website: courses.ctda.hcmus.edu.vn – Moodle

Course's objectives

- basic programming concepts
- basic programming structures
- well-organized C++ programs to solve basic problems

Textbooks

- *Tony Gaddis*, Starting out with C++ From Control Structures through Objects, Pearson, 8th edition, 2015

- *D. S. Malik*, C++ Programming: From Problem Analysis to Program Design (more exercises)
- *Vũ Quốc Hoàng*, Bí kíp luyện Lập trình cơ bản với C https://github.com/vqhBook/C
- *Trần Đan Thư*, Giáo trình Nhập môn lập trình Khoa CNTT – Trường ĐHKHTN Tp.HCM, 2011

Course's content

W	Topic	Textbook
1	Introduction to Programming	1.1-1.7
2	Basic elements	2.1-2.17, 3.1-3.9
3	Flowchart	2.4 (VQHoang book)
	Control flow statements: If-else	4.1-4.14
4	Control flow statements: Loop	5.1-5.10, 5.12
5	Functions	6.1-6.13
6	Midterm	
7	Array	7.1-7.8
8	2D Array	7.9-7.11
	String	10.1-10.8
9	Struct	11.1-11.8
10	File	5.11, 12.1-12.10

Grading policy

lab	40 points
midterm	25 points
final	35 points
bonus	∞
Total	100 points

bonus:

- in-class exercises
- quizzes:
  $+1$ if $sum([Qi, i = 1..n]) >= n/2$
  $+1$ for each Qi really "$>$"

Q1	Q2	Q3	Q4	Q5	...
T	F	T	T	T	
		$+1$	$+1$	$+1$	

Regulations

- Students who are absent for more than 3 theory sessions are not allowed to take the exams
- Students whose lab's score is lower than 10/40 points will be graded 0 for the final and will fail this course
- For any kind of cheating and plagiarism, students will be graded 0 for the course.

Cheating:

* Sharing code: by copying, retyping, looking at, or supplying a file
* Describing: verbal description of code from one person to another.
* Coaching: helping your friend to write a lab, line by line
* Searching the Web for solutions
* Copying code from a previous course or online solution

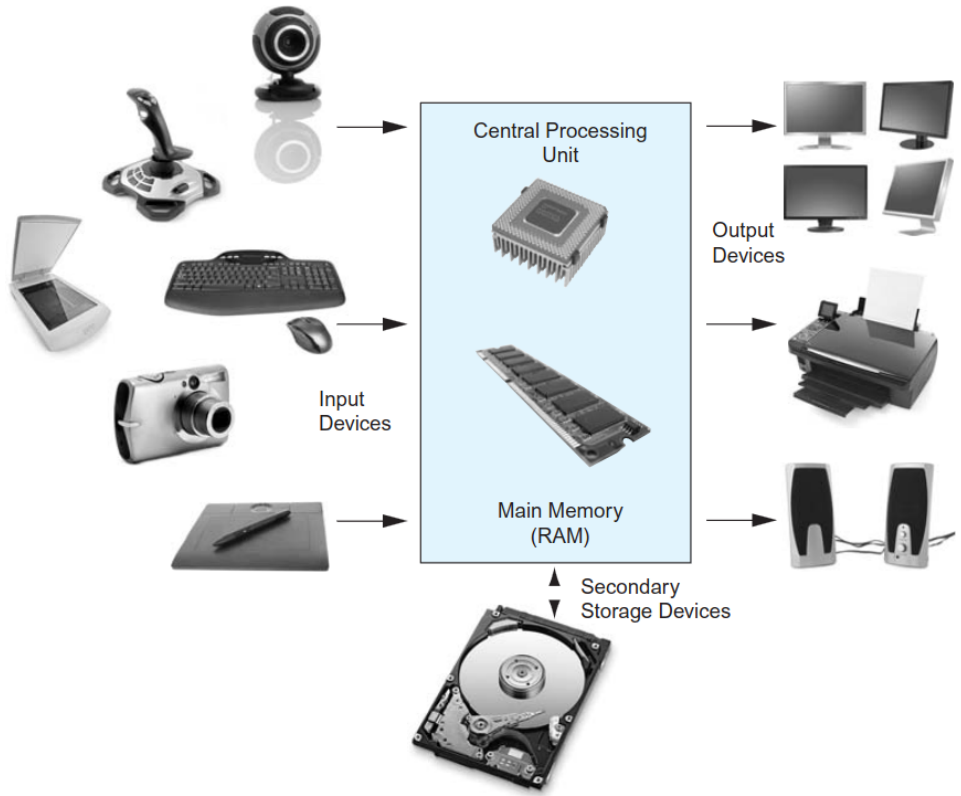
(CMU cheating description)

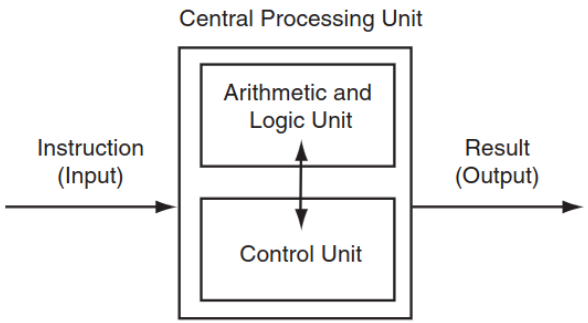
Gettting help

- Moodle: courses.ctda.hcmus.edu.vn
- Email: tttnhi@mso.hcmus.edu.vn
  ptpuyen@fit.hcmus.edu.vn
  nvqhuy@fit.hcmus.edu.vn
  ex: [CSC10001] Ask for additional materials
- Office hours: I63, Monday/Wednesday/Friday
- Discord: https://discord.gg/URmpKjQN

Getting start: Computer Systems

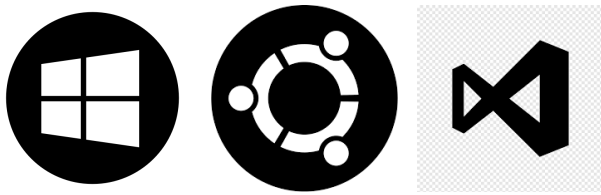
Hardware





Software

System Software



Application Software

Why do programming?

Human language

```
Add 5 and 8
```

Programming language

Low-level (Assembly)

```
add 0x8(%ebp),%eax
//instruction
```

High-level (C++, Python, Java, ...)

```
int a = 5 + 8;
//instruction / statement
```

Machine language

```
011000110101
//instruction
```

⇓

each different type of CPU has its own machine language

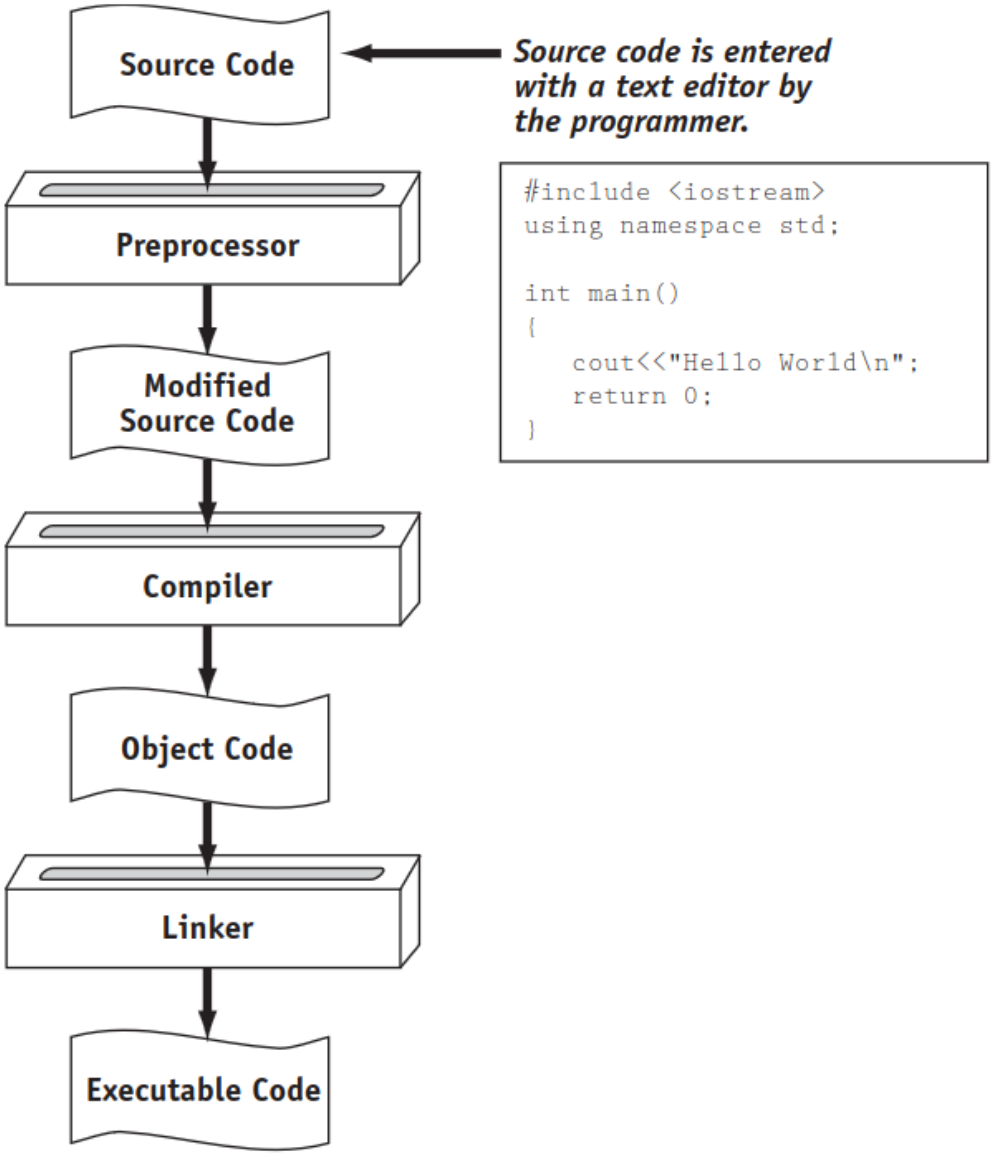
→ A program is a *set of instructions* a computer follows in order to perform a task

A program

```
1  //This program calculates the user's age
2  #include <iostream>
3  using namespace std;
4
5  int main(){
6      int year, age;
7      // Get the year born
8      cout << "When were you born? ";
9      cin >> year;
10     // Calculate the age
11     age = 2023 - year;
12     // Display the age
13     cout << "You are " << age << endl;
14     return 0;
15 }
```

statements (typed into a computer by a **text editor**) $\rightarrow$ **source code** $\rightarrow$ saved to a file $=$ **source file**

Source Code, Object Code, and Executable Code



Preprocessor executes lines beginning with # symbol (preprocessor directive)

Compiler translates source code instructions into machine language instructions

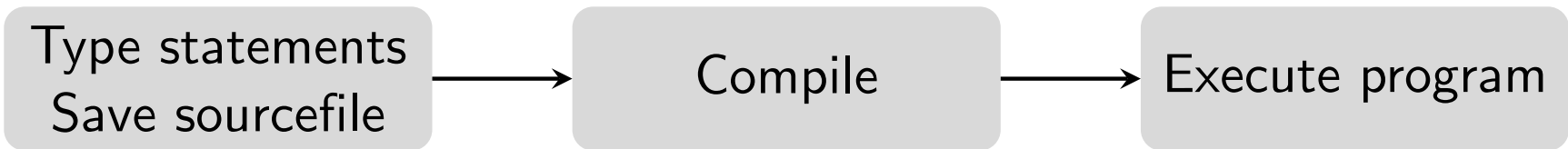
Linker combines object code with the necessary library

```
>> g++ hello.cpp -o hello(
```

Integrated Development Environments (IDEs)

- include: text editor, compiler, debugger, ...
- single click (F5, F11, ...)

Basic Program Development



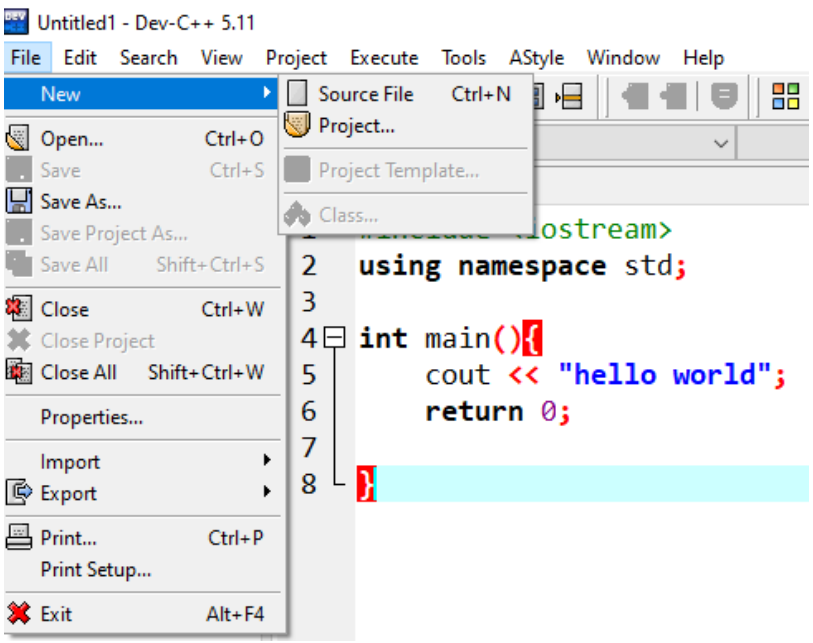
What do we need?

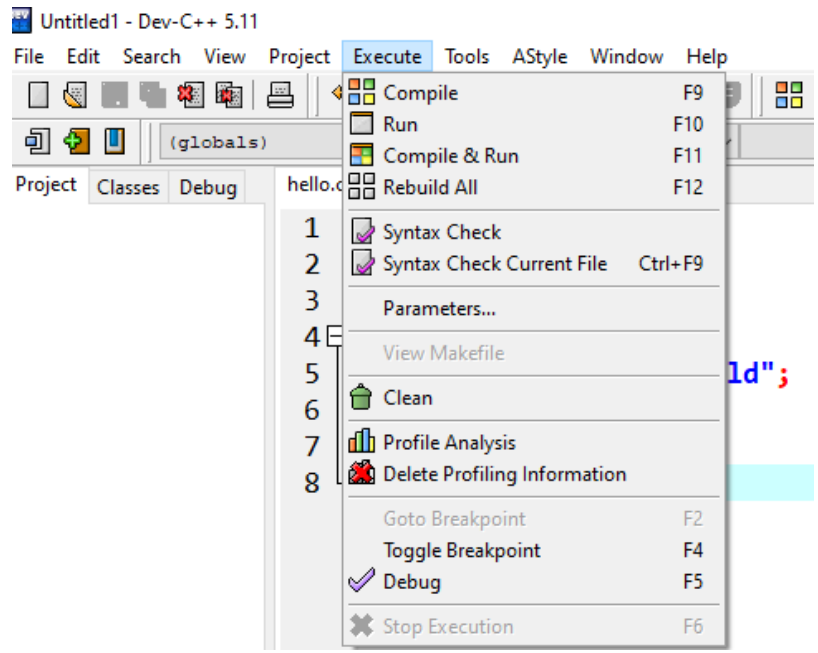
Hardware: a computer
Software:

- a text editor + a compiler (C++), or
- IDE

```
>> g++ hello.cpp -o hello(.exe)
```

```
[compile and run] F5, F11 ...
```





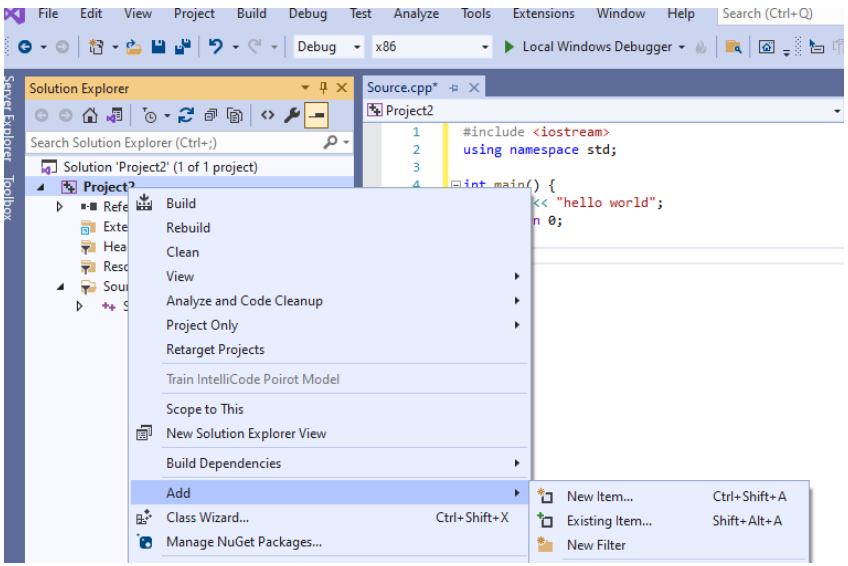
What do we need?

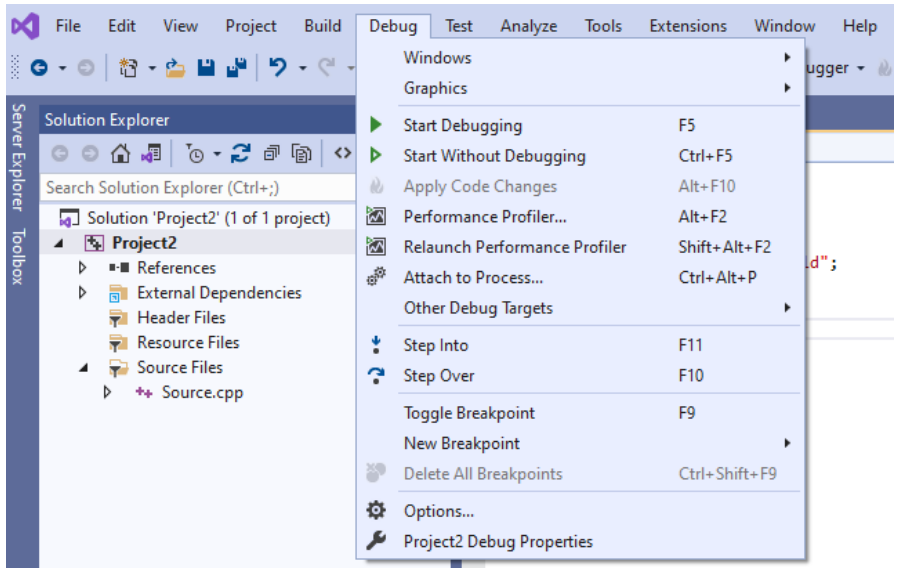
Hardware: a computer
Software:

- a text editor + a compiler (C++), or
- IDE

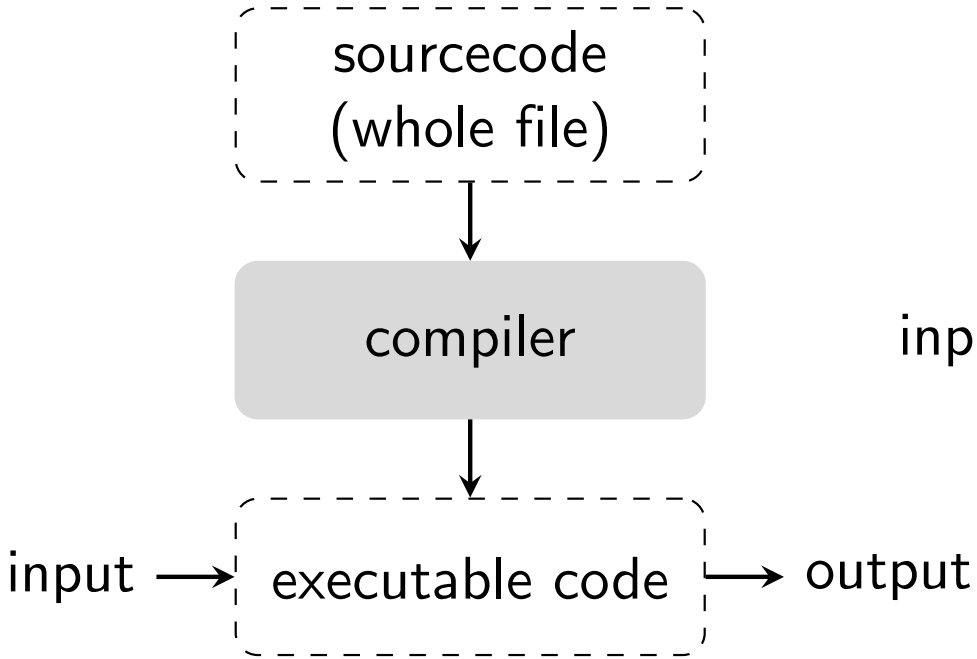
```
>> g++ hello.cpp -o hello(.exe)
```

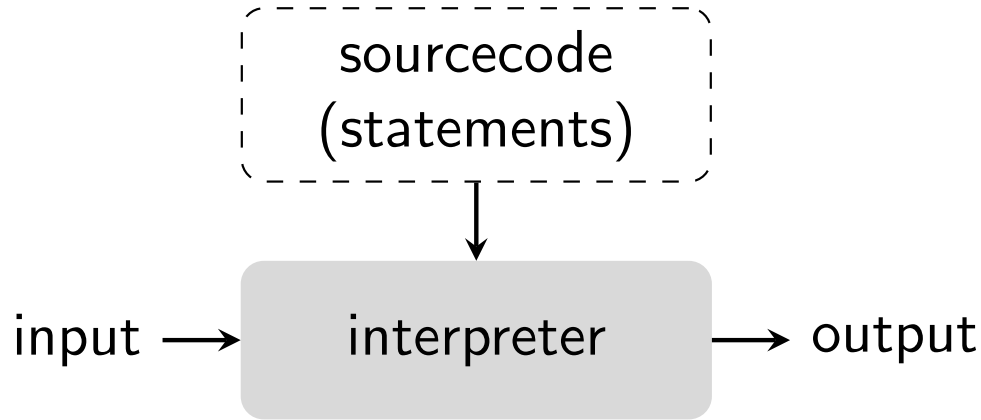
```
[compile and run] F5, F11 ...
```





Compilation and Interpretation





`C, C++, Scala, Java, ...`

`Python, Ruby, PHP, Java, ...`

Program Paradigms

```
1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int year, age;
6      // Get the year born
7      cout << "When were you born? ";
8      cin >> year;
9      // Calculate the age
10     age = 2023 - year;
11     // Display the age
12     cout << "You are " << age << endl;
13     return 0;
14 }
```

Key words: `include`, `using`, `namespace`, `int`, ...

Identifiers: `main`, `year`, `age`

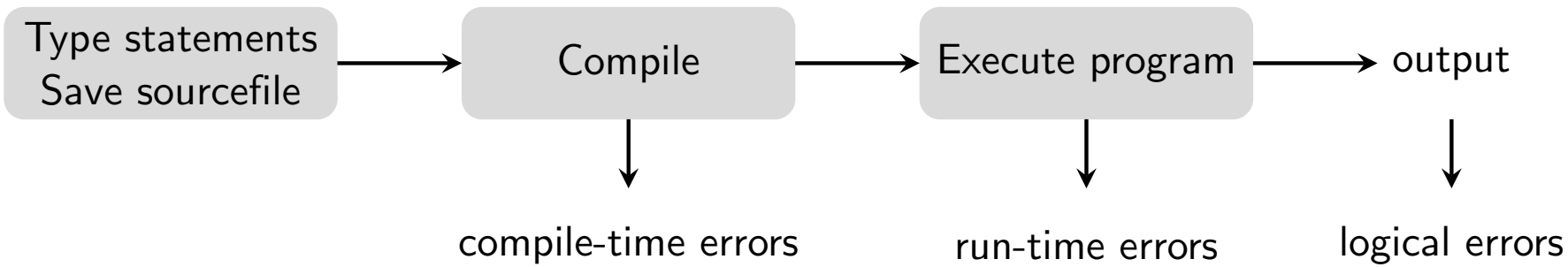
Operators: `-`, `=`

Punctuation: `;`

Syntax

Bugs (Errors)

Bug: a mistake in a program $\rightarrow$ find and fix bugs "**debug**"

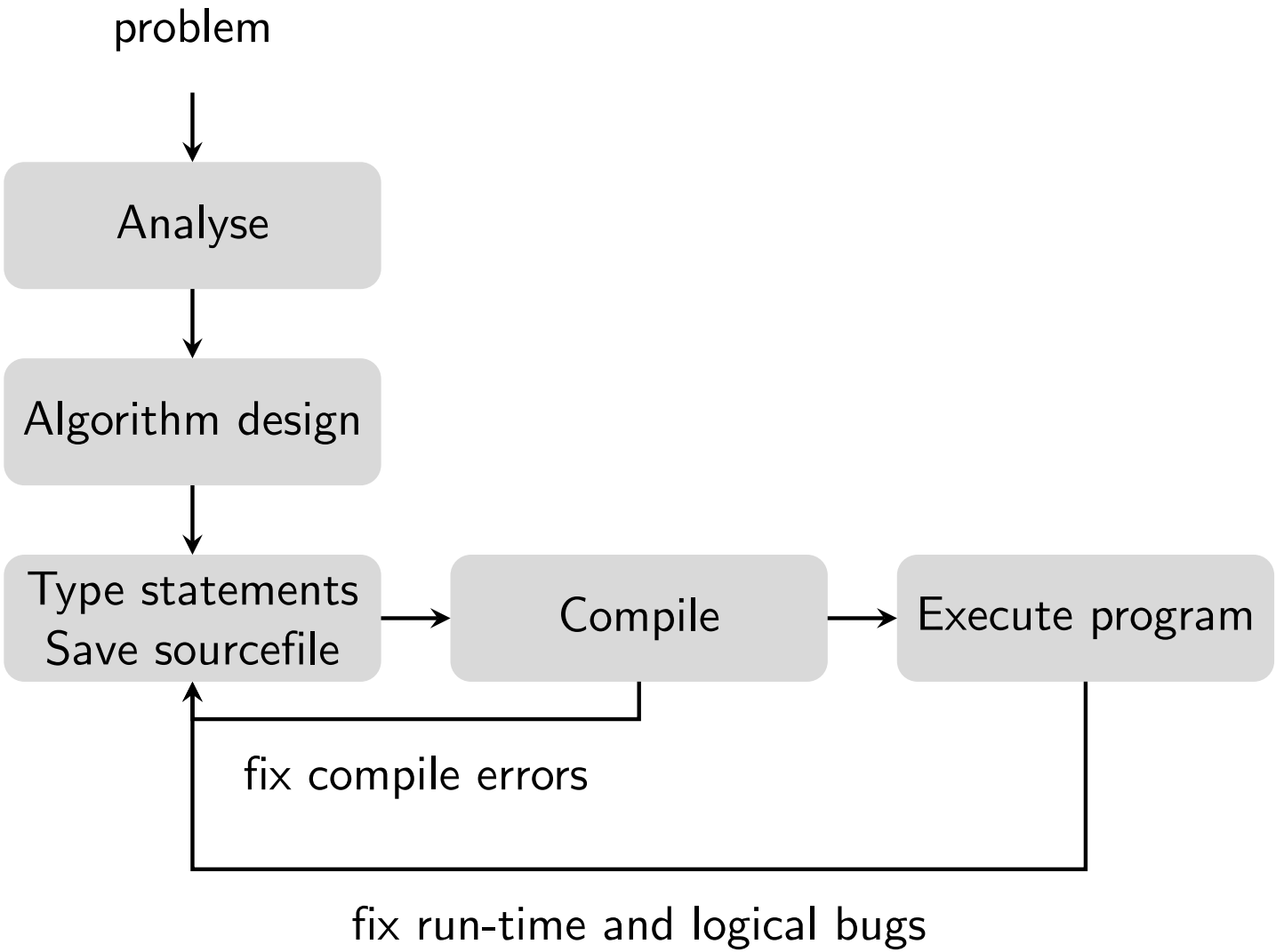



```
int year, age
```
(syntax error)

```
age = 2023 / age;
```

```
cin >> age;
```

Basic Program Development



Exercises

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

int main(){
    int year, age;
    // Get the year born
    cout << "When were you born? ";
    cin >> year;
    // Calculate the age
    age = 2023 - year;
    // Display the age
    cout << "You are " << age << endl;
    return 0;
}
```

Write a program to

- introduce yourself
- welcome a new student

TODO

- Finish chapter 1
- Read chapter 2, 3