Computer Science 1: IT and web

coefficient: 2
credit: 3
continuous control weight: 50%
exam weight: 50%

Mrs HamzaCherif Souaad

Course 5: Programming language

Goals

- What is a programming language?

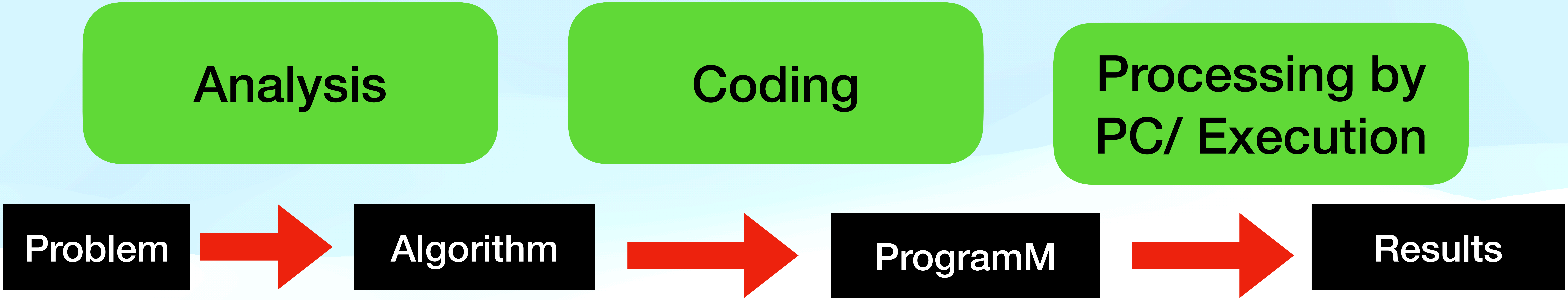
1-Introduction

1. Introduction

- Just as humans have designed languages to communicate with each other, programmers have created a series of languages to communicate with computers so that they can process information automatically.

- Generally speaking, we can say that the primary objective of automatic information processing is to ask a computer to process information in an automatic manner and present the results according to the plan that has been made.

1. Introduction



Analysis

Coding

Processing by PC/ Execution

Problem

Algorithm

ProgramM

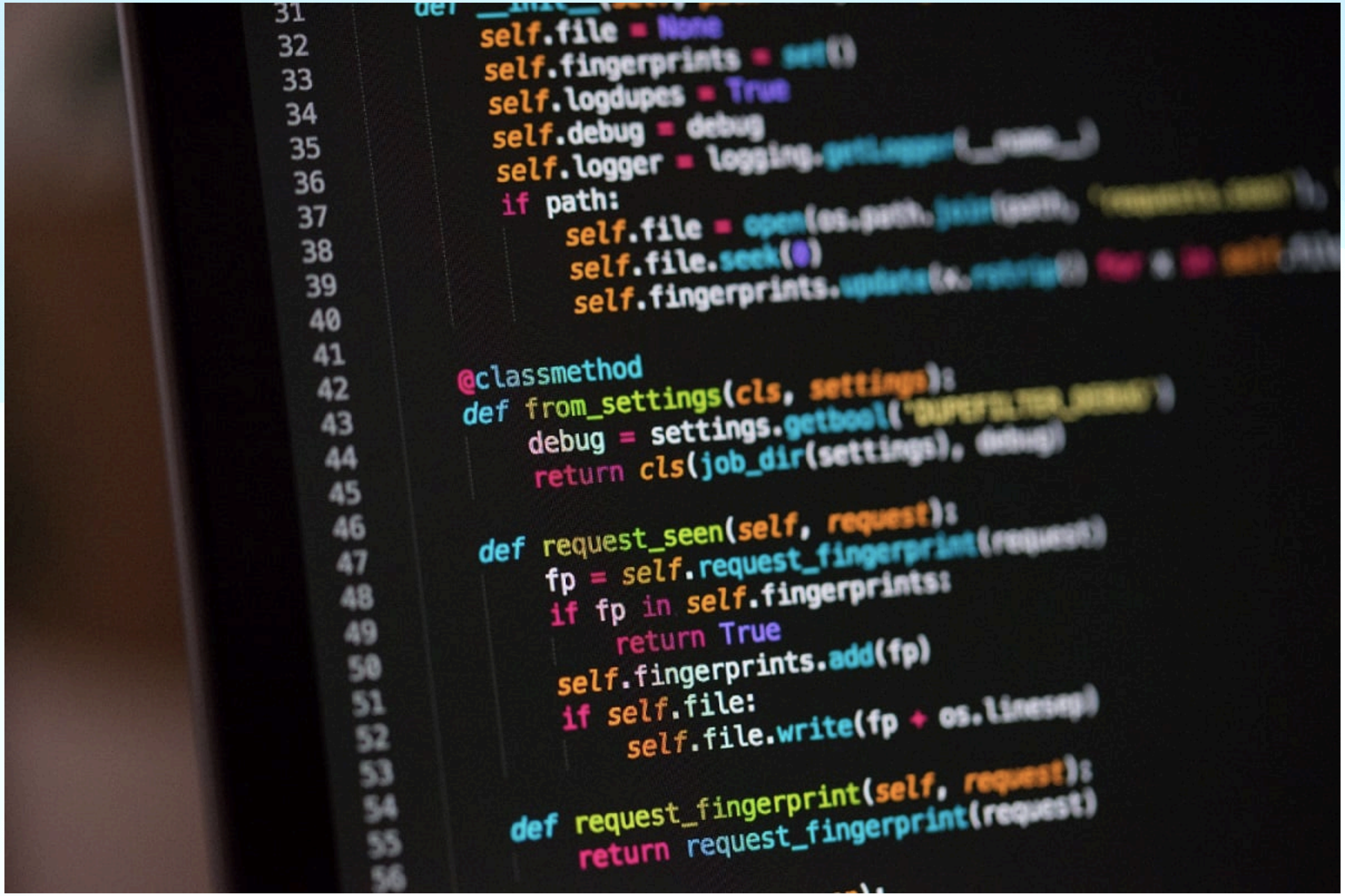
Results

2-Definition

2. Definition

- A programming language is a conventional notation intended for formulating algorithms and producing computer programs that apply them.
- Similar to a natural language, a programming language is made up of an alphabet, vocabulary, grammar rules, and meanings

1. Introduction



```
    def __init__(self,
        self.file = None
        self.fingerprints = set()
        self.logdupes = True
        self.debug = debug
        self.logger = logging.getLogger(__name__)
        if path:
            self.file = open(os.path.join(path, 'requests.seen'), '
            self.file.seek(0)
            self.fingerprints.update(x.rstrip() for x in self.fi

    @classmethod
    def from_settings(cls, settings):
        debug = settings.getbool('DUPEFILTER_DEBUG')
        return cls(job_dir(settings), debug)

    def request_seen(self, request):
        fp = self.request_fingerprint(request)
        if fp in self.fingerprints:
            return True
        self.fingerprints.add(fp)
        if self.file:
            self.file.write(fp + os.linesep)

    def request_fingerprint(self, request):
        return request_fingerprint(request)
```

2. Definition

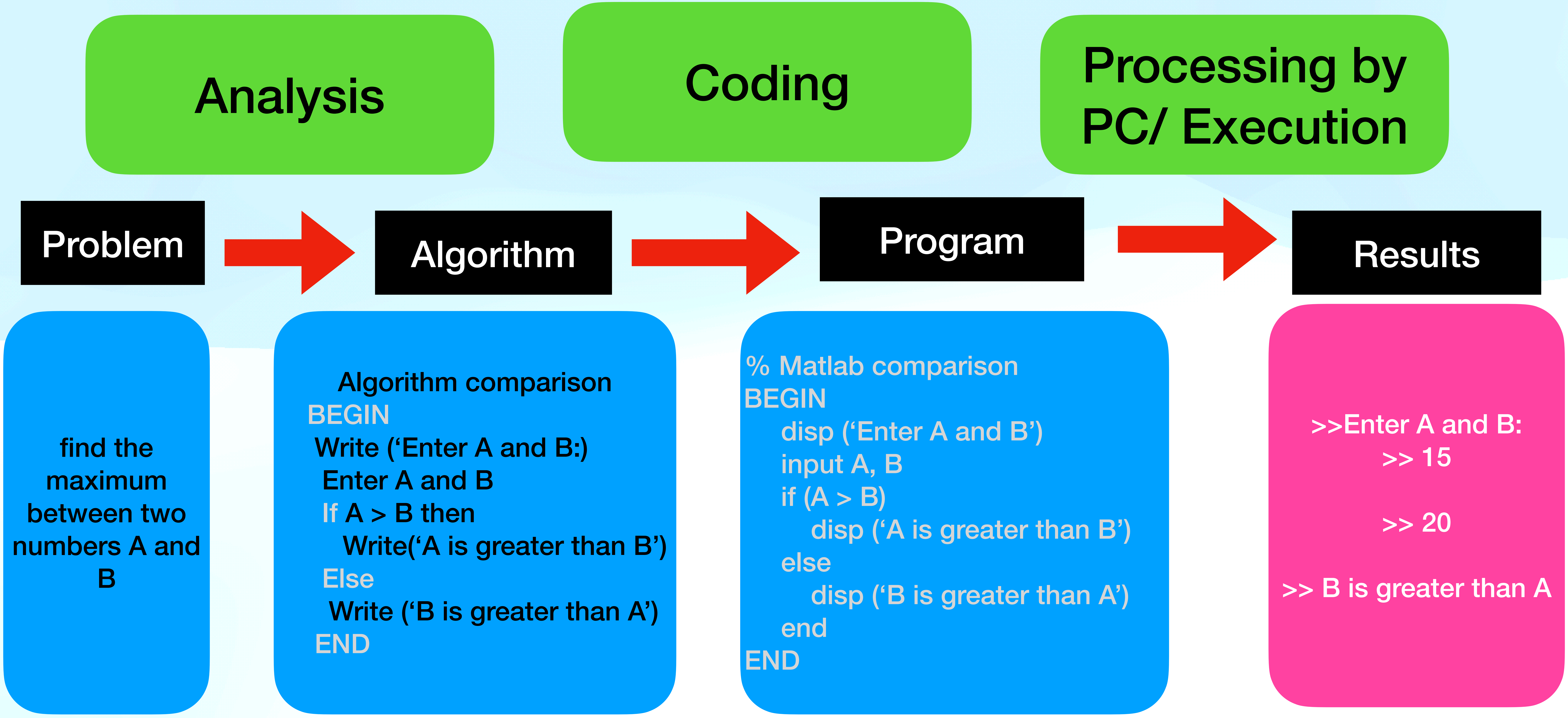
- An algorithm is a step-by-step procedure. It is a set of rules to follow to accomplish a task or solve a problem.

- Long before the emergence of computing, humans were in fact already using algorithms. We can consider that cooking recipes, mathematical operations or even instructions for assembling a piece of furniture are algorithms.

- In computer programming, algorithms are sets of rules that tell the computer how to perform a task.

- An algorithm can be translated into any programming language.

1. Introduction



Analysis

Coding

Processing by PC/ Execution

Problem → Algorithm → Program → Results

find the maximum between two numbers A and B

```
   Algorithm comparison
BEGIN
 Write (‘Enter A and B:)
  Enter A and B
  If A > B then
     Write(‘A is greater than B’)
  Else
   Write (‘B is greater than A’)
 END
```

```
% Matlab comparison
BEGIN
    disp (‘Enter A and B’)
    input A, B
    if (A > B)
        disp (‘A is greater than B’)
    else
        disp (‘B is greater than A’)
    end
END
```

```
>>Enter A and B:
>> 15

>> 20

>> B is greater than A
```

3-Types of languages

3. Types of languages

- There are over 2,500 programming languages. They are classified according to their level of abstraction, from machine language to human logic language. Types of programming languages are 2 types: low level and high level

- A low-level language means that it is a language that is very close to the level of electronic components, while a high-level language is a language more easily understood by humans (developers), and that it will have to be transformed to be understood by the machine.

3. Types of languages

- The choice of one language or another should only be made according to the project, according to the constraints, the use, etc.

- If you need to control electronics directly then it will be a low-level language (C/C++). Same if you are at the operating system level and you need great performance (C/C++/Rust) or in video games, data processing, etc... then same thing (with C# for example).

- If, on the other hand, your application must run on hundreds of different machines, on different OSes, then we will use a slightly higher level language (Java, Python, Javascript, etc.).

3. Types of languages

- The choice of one language or another should only be made according to the project, according to the constraints, the use, etc.

- If you need to control electronics directly then it will be a low-level language (C/C++). Same if you are at the operating system level and you need great performance (C/C++/Rust) or in video games, data processing, etc... then same thing (with C# for example).

- If, on the other hand, your application must run on hundreds of different machines, on different OSes, then we will use a slightly higher level language (Java, Python, Javascript, etc.).

3. Types of languages

- High-level languages can be divided into several families based on their programming paradigm:
    - **Imperative:** Imperative programming is the most common type of programming. It consists of listing a sequence of orders to the computer. The code will then be executed line by line until the end of its execution.

3. Types of languages

- **Functional:** these languages are based on the mathematical concept of function. Functional programming is primarily characterized by the ability to declare and call functions inside other functions.

3. Types of languages

- **Logic:** Logic programming is generally used in artificial intelligence. This is programming based on the calculation of predicates. Here, statement is a clause that describes the relationship between data. Among these languages, we can cite Prolog and Mercury.

3. Types of languages

- **Object Oriented Programming (OOP):** is a paradigm in which we create and then use objects. An object is like a structure that represents an entity. We can then add methods and characteristics to these objects.

4-Evolution of langage

4. Evolution of langage

- **1843 – Ada Lovelace program for Charles Babbage's Analytical Engine:** Considered the "father of the computer", Charles Babbage proposed the analytical machine, a general-purpose mechanical computer, in 1837. In 1843, Ada Lovelace, the world's first female computer programmer, was the first to realize that an analytical engine could do more than simple calculations and published the first algorithms designed to work on it.

4. Evolution of langage

- **1949 – Assembly** becomes the first widely used type of programming language. It was the first to allow programmers to write code using symbols instead of binary code.
- **1952 – Autocode** becomes the first compiled computer programming language. Originally developed by Alick Glennie for the Mark 1 computer, autocode is a term used to refer to a family of "simplified coding systems".

4. Evolution of langage

- **1957 – Fortran** is designed for numerical computing and scientific computing. Developed by John Backus and IBM, Fortran is considered the first high-level language, designed to be relatively easy to learn and widely applicable.
- The software that powered NASA's Voyager 1 and 2 spacecraft was written in Fortran 5.

4. Evolution of langage

- **1959 – Cobol establishes itself as a common business-oriented language**
- It has become the language of choice for professional applications and owes much of its popularity to IBM. In 1997, 80% of businesses worldwide were using Cobol. In 2020, the U.S. Internal Revenue Service transitioned to a Cobol-based Individual Master File (IMF) to help citizens and businesses with relief and coronavirus pandemic relief programs.

4. Evolution of langage

- **1964 – BASIC (Beginners' All-Purpose Symbolic Instruction Code)**
- Designed by John G. Kennedy and Thomas E. Kurtz, the BASIC programming language was intended to be as accessible as possible, and it is no surprise that it came to dominate the home computer market that emerged in the 1970s In 1991, Microsoft developed Visual Basic by combining an updated version of BASIC with a visual forms builder.

4. Evolution of langage

- **1972 – The C language begins to shape the future of the personal computer.**
- It is the mother of almost all high-level languages and remains one of the most popular languages in the world today. It can be used in computer operating systems, embedded system implementations, and web environments using the Common Gateway Interface (CGI).
- Programming languages derived from C include C#, D, Go, Java, JavaScript, Limbo, Python and others.

5-Current trend

5.Current trend

- **Python**
- Developed by Guido van Rossum in the 1990s, Python is one of the most popular programming languages in the world. Its continued success is due to its ease of use, which allows beginners to quickly learn to write simple programs.
- Python is primarily used to create programs that use artificial intelligence, data analysis, and machine learning.

5.Current trend

- **Javascript**
- Created in 1995 by Netscape, this programming language is widely used with HTML and CSS (which are markup languages and not programming languages) to create interactive websites and various types of applications. Javascript revolutionized web pages, which were very static at the time, by adding dynamic animations and increasing the possibilities for interaction with the user.

5.Current trend

- **Java**
- It is the primary programming language used to create enterprise applications. Created in 1995, it is a so-called object-oriented programming language which allows you to define software objects that can interact with each other through the exchange of messages.
- Java was created as a compiled language: this means that an application written in this language can be executed by any platform supporting Java (according to the motto “write once, run anywhere”).

5.Current trend

- **C/C++**
- These two languages are used to create applications and platforms for the IoT (Internet of Things) and systems that host applications. Their learning will allow you to acquire the basics of programming and a better understanding of other languages.

5.Current trend

- **C#**
- Developed by Microsoft to speed up the C language (as a competitor to Java), it was incorporated into the company's .NET framework and is used for video game development, server-side programming, and mobile applications for Windows . Its syntax is similar to that of C, C+ and Java.

5.Current trend

- **PHP**
- Developed in the mid-1990s, PHP is a server-side scripting language intended for developing dynamic and interactive websites. Used by many developers around the world, it benefits from a large community which has produced free software and frameworks making it easier to use this language.