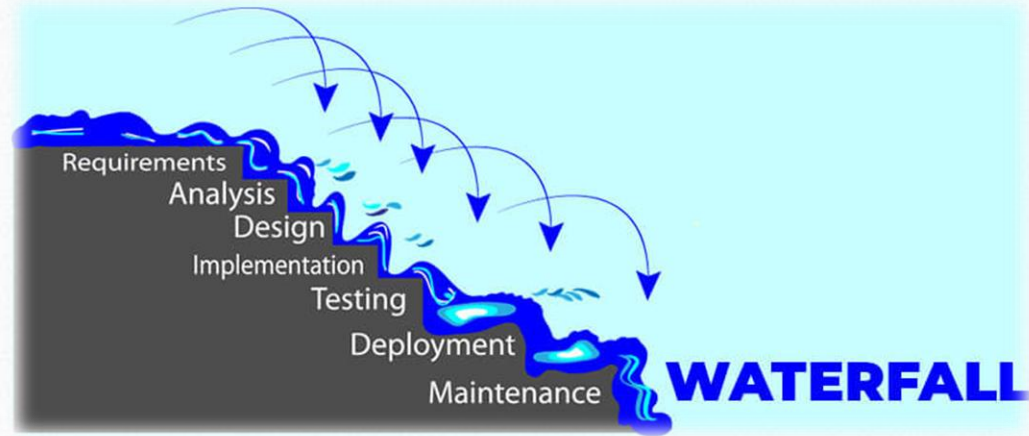


Software Engineering

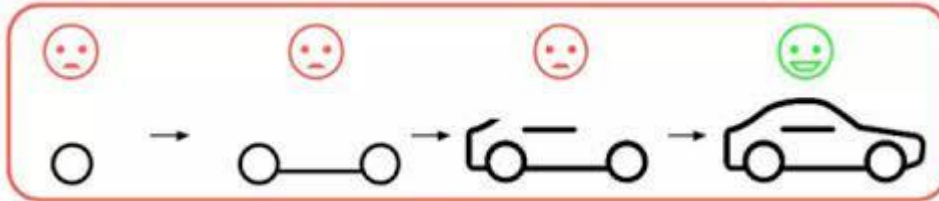
Software Process Models

Topics covered

- Waterfall model



- Incremental model *Mô hình tăng dần*



Example of the software process

- Problems: Write a console program to solve $ax+b=0$

1. Requirements definition/Specification (**what**)

Input a,b. Output the result

2. Design (**how**): the solution to implement

When $a \neq 0$: the equation has one solution $x = -b / a$.

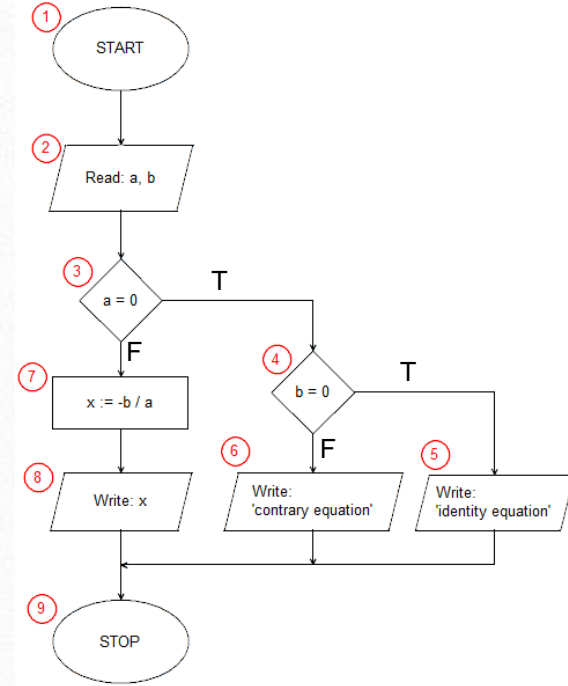
When $a = 0$:

- $b = 0$: infinite solutions
- $b \neq 0$: no solutions

3. Implementation (**do**): using C/C++ language to write the program

4. Testing (**check**): run program

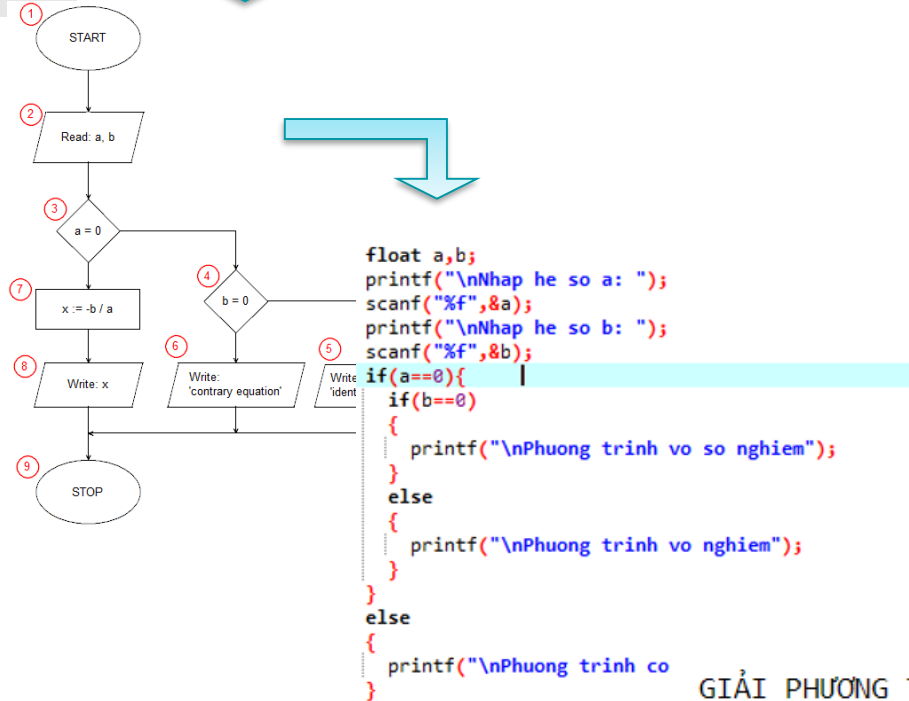
- Case 1: $a \neq 0$
- Case 2: $a=0$ & $b=0$
- Case 3: $a=0$ & $b \neq 0$



Requirements:

- Input: a,b
- Output: the result

Process 1



GIẢI PHƯƠNG TRÌNH $ax + b = 0$

-
1. Giải phương trình với $a < > 0$
 2. Giải phương trình với $a=0$ và $b < > 0$
 3. Giải phương trình với $a=0$ và $b=0$
- Mời chọn (1-3):

Process 2

Requirements:

- Input: a,b
- Output: the result

Design, Implementation, Testing

GIẢI PHƯƠNG TRÌNH $ax + b = 0$

1. Giải phương trình với $a \neq 0$
- Mời chọn (1):

Requirements:

- Input: a,b
- Output: the result

Design, Implementation, Testing

GIẢI PHƯƠNG TRÌNH $ax + b = 0$

1. Giải phương trình với $a \neq 0$
 2. Giải phương trình với $a=0$ và $b \neq 0$
- Mời chọn (1-2):

Requirements:

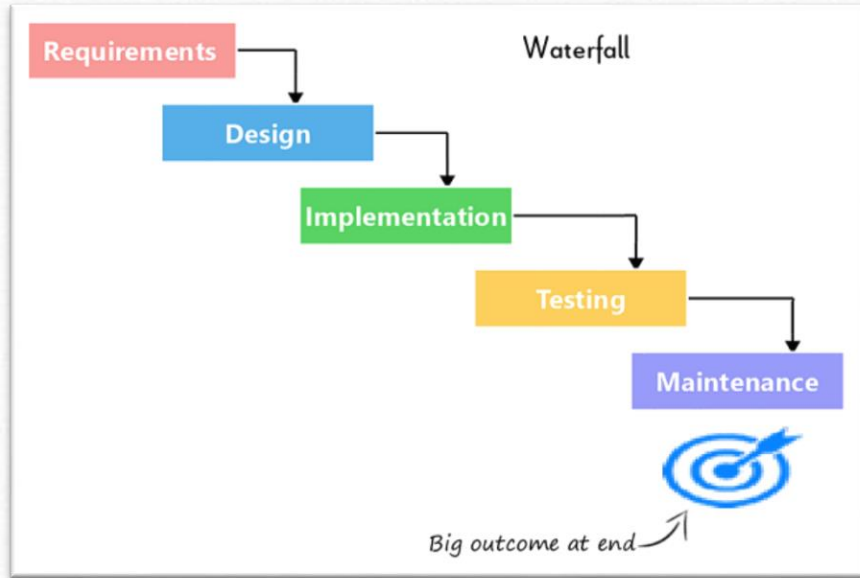
- Input: a,b
- Output: the result

Design, Implementation, Testing

GIẢI PHƯƠNG TRÌNH $ax + b = 0$

1. Giải phương trình với $a \neq 0$
 2. Giải phương trình với $a=0$ và $b \neq 0$
 3. Giải phương trình với $a=0$ và $b=0$
- Mời chọn (1-3):

Waterfall Model



- Waterfall model is a process of software development where...
 - A phase has to be **complete before moving** into the next phase with *no overlap between the phases*
 - The software team is **not allowed to return** to the previous phase.
 - The **big outcome** is delivered

Waterfall model problems

Advantages

- Easy to use
- Easy to follow

Disadvantages

- No parallelism
- No feedback
- Difficult to respond to changing customer requirements



Waterfall design

Waterfall deliver



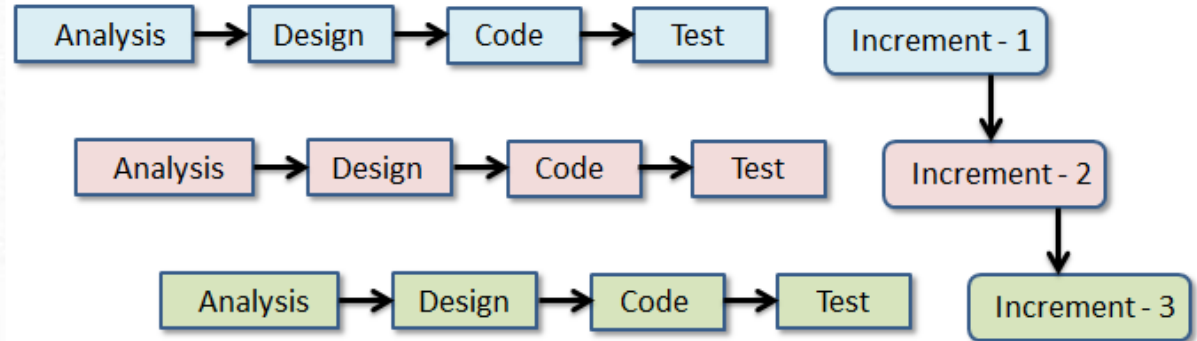
When to use the waterfall model?

- Requirement is clear
- Requirement is not changing frequently
- Environment is stable

.

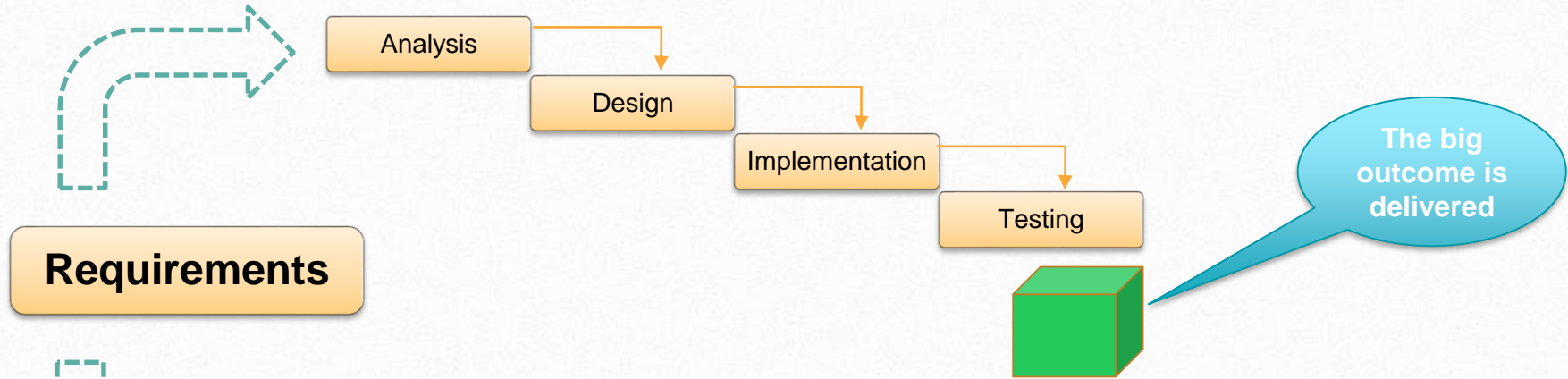
Incremental Model

- The software is divided into separate **increments (modules, components)**
- Each increment done through the *requirements, design, coding and testing* phases
- When any increment is ready, then the **increment is delivered** to the customer
- The increments are delivered to the customer one by one by **integrating new increment** with old one

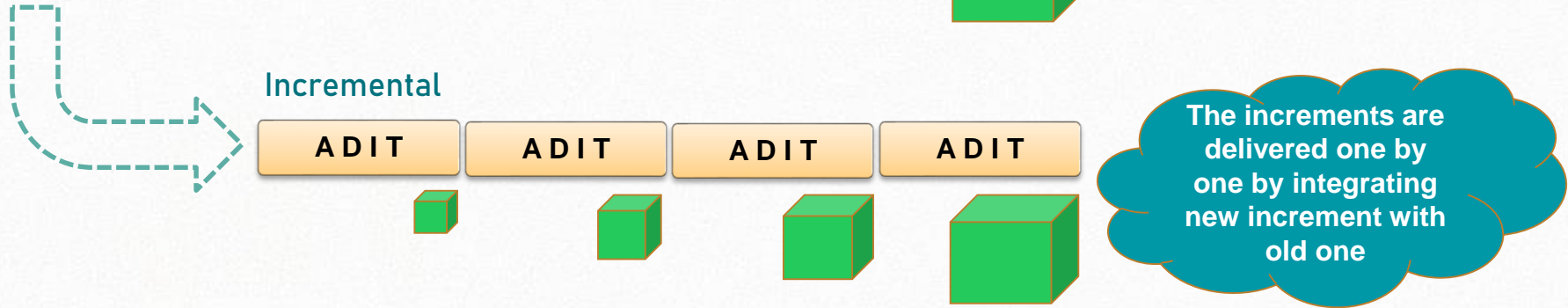


Incremental development

Waterfall



Incremental



Incremental development benefits

- Give **rapid delivery**
 - Customers are able to use the software earlier than waterfall process.
- It is easier to get **customer feedback** on the development work that has been done.
- The **cost** of accommodating changing customer requirements is **reduced**.

Incremental development problems

- **System structure** tends to degrade as new increments are added.
- Needs a **proper design** to integrate the components

When to use the incremental model?

- When major **requirements** are understood but *some requirements can evolve within the passage of time.*
- When demand for an **early release** of a product arises
- When a **customer** has no problem with the budget but he demands more and more quality in software.

A case study of incremental model (1)

- The MHC-PMS (Mental Health Care-Patient Management System) is a patient information system to support mental health care that is intended for use in clinics.
- MHC-PMS key features
 - **create** records for patients,
 - **edit** the information in the system,
 - **view** patient history
 - **report** data summaries so that doctors can quickly learn about the key problems and treatments that have been prescribed.

A case study of incremental model (2)

- According to the incremental model, the system is divided into three increments
 - Increment 1: Create and View
 - Increment 2: Edit
 - Increment 3: Report
- The increment 1 undergoes the phases of *requirements gathering and analysis, design, implementation and testing*. When this increment is ready, this one is delivered to the customer.
- After that, increment 2
- After that, increment 3
- Result: one system is produced and delivered to the customer in increments.

Key points

- Software processes are the activities involved in producing a software system.
 1. requirements,
 2. design,
 3. implementation,
 4. testing and
 5. evolution/maintenance
- In the **waterfall model**, they are organized in **sequence**
- In **incremental development** they are **inter-leaved**

Plan-driven and agile processes

- **Plan-driven** processes are processes where all of the process activities are planned in advance and progress is measured against this plan.
- In **agile** processes, planning is incremental and it is easier to change the process to reflect changing customer requirements.
- In practice, most practical processes include elements of **both plan-driven and agile approaches**.
- There are **no right or wrong** software processes.

Assignments

A case study of software process models

- Summary the software project (project name, short description)
- **Describe the process** to develop project using waterfall model and incremental model