

12.1 Program Development Life cycle

Name: _____ Class: _____ Date: _____

Total: 18 marks

KEY WORDS development life cycle • maintenance • waterfall
• iterative model • adaptive maintenance

Objective

Build the skills to answer exam questions on the **program development life cycle**
– its stages and the common models.

You must be able to:

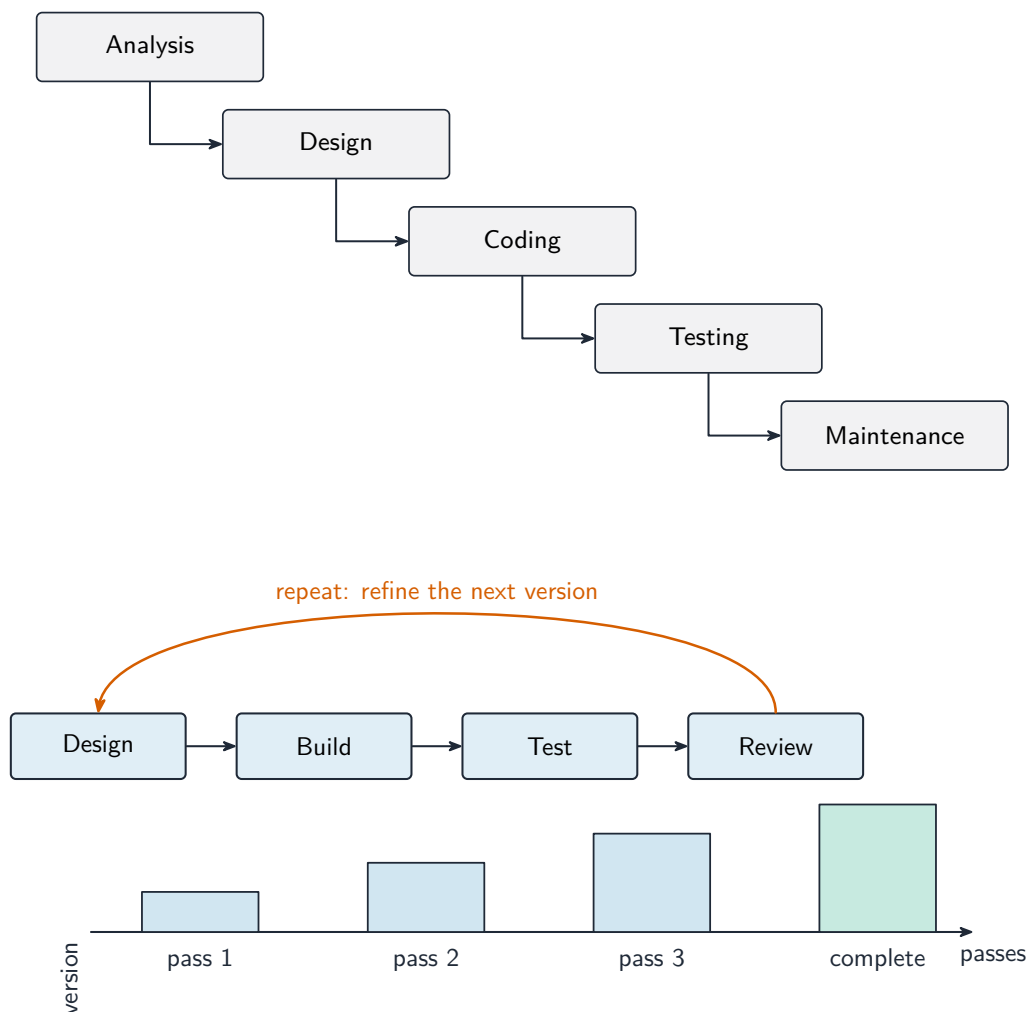
- explain the **purpose** of a development life cycle
- describe the **analysis, design, coding, testing, maintenance** stages
- compare models (**waterfall, iterative, RAD, agile**) and choose one

1 Key knowledge and worked examples

Study these first. Each one shows the method for a question type used later – follow the steps and you can do the Practice and Exam-style questions yourself.

■ Purpose and the five stages

A development life cycle is the set of stages from idea to maintained software. It exists to **plan, manage and control** a project – to build the right product, on time, with good quality.



The iterative model contrasts with the waterfall

- **analysis** – find **what** the program must do (gather requirements).
- **design** – decide **how**: data structures, algorithms, modules, interface.
- **coding** – write the source code from the design.
- **testing** – run against test data and fix bugs.
- **maintenance** – after release, keep it working and useful.

Three kinds of **maintenance**: **corrective** (fix bugs found in use), **adaptive** (change it for a new operating system, new law or new hardware), and **perfective** (improve performance or add features users ask for).

■ The models

Model	Principle	Best when	Drawback
Waterfall	linear; finish each stage first	requirements are stable	poor at mid-project change
Iterative	repeated passes refine it	requirements emerge over time	harder to estimate
RAD	fast prototypes + user feedback	requirements change	needs user availability
Agile	short sprints, constant testing	change is constant	needs a committed customer

Choose by how **clear** and **stable** the requirements are at the start.

■ Key knowledge: testing and maintenance within the life cycle

Alpha testing is internal; **beta** testing uses selected external users; **acceptance** testing is the customer's check that agreed requirements are met. A walkthrough follows the design/code with others to find errors. Choose normal, abnormal and boundary values and compare actual with expected results.

After release, **corrective** maintenance fixes faults, **adaptive** responds to a changed environment such as new hardware, and **perfective** improves performance or usability. Testing and amendments can occur repeatedly, not only once at the end.

2 Practice

Now apply the methods above.

2.1 State the **purpose** of a development life cycle. [1]

development life cycle

2.2 Name the **five** stages of the life cycle, **in order**. [2]



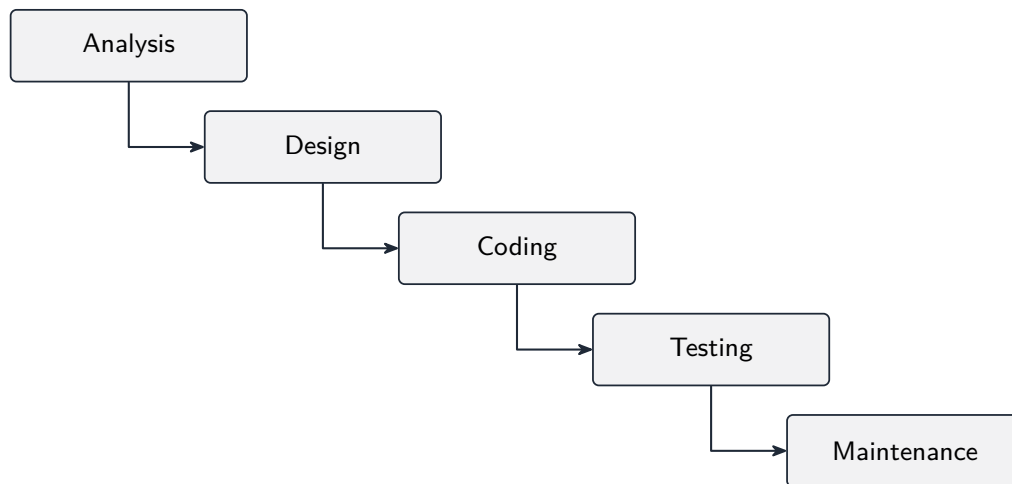
2.3 At which stage are the **data structures and algorithms** decided? [1]

data structures and algorithms

2.4 State what happens during the **maintenance** stage. [1]

maintenance

2.5 A customer checks a delivered program against the agreed requirements. Name this testing stage. A developer then modifies the program to run on new hardware: name the maintenance type. [2]



3 Exam-style questions

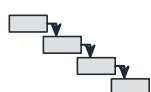
3.1 Describe what happens in the **analysis** stage and in the **testing** stage. [4]

analysis

testing

3.2 State **one** benefit of the **waterfall** model and **one** benefit of the **iterative** model. [2]

waterfall



iterative



3.3 A company's project has requirements that are likely to **change a lot** during development. State a suitable life-cycle model and **justify** your choice. [3]



requirements

3.4 (a) State what **adaptive** maintenance is. [1]

adaptive maintenance

(b) Give **one** example of a change that would need adaptive maintenance. [1]
