

4.2 Types of programming language, translators and IDEs

Name: _____ Class: _____ Date: _____ Total: 25 marks

KEY WORDS high-level language 高级语言 • machine code 机器码 • interpreter 解释器
• low-level language 低级语言 • compiler vs interpreter 编译器

Objective

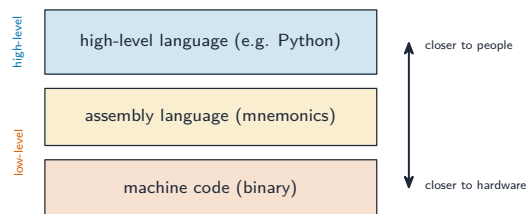
Build the skills to answer exam questions on **programming languages, translators and IDEs** 编程语言、翻译程序与集成开发环境 — **high/low-level** 高级/低级 languages, **compiler vs interpreter** 编译器/解释器, and IDE tools.

You must be able to:

- compare high-level and low-level languages
- compare a compiler and an interpreter
- list the tools in an IDE

1 Worked examples

■ Translators



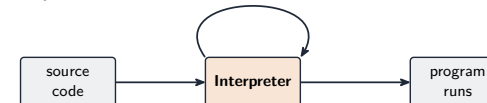
High-level: easy, runs on any computer. Low-level (assembly, machine code): tied to one CPU

Compiler — translate the whole program once



translated once; the machine code then runs many times

Interpreter — one line at a time



reads, translates and runs one line at a time (every run); stops at the first error

Compiler: translates all at once, finished program runs fast. Interpreter: line by line, stops at first error

■ The two internet-security measures the syllabus names

A **firewall** 防火墙 sits between the network and the outside. It examines the traffic entering and leaving, compares it with a set of criteria, blocks anything that fails, keeps a log of the traffic, and can block named sites or IP addresses. It can be software, hardware, or both. It cannot stop a threat carried in by an authorised user on a memory stick, and it cannot help if a user gives their password away.

A **proxy server** 代理服务器 sits between the user and the internet and passes requests on. The web server sees the proxy's address, not the user's, so it hides the user's identity; it can filter traffic and block unwanted sites; it keeps a **cache** of pages so a repeated request is served quickly; and it can act as a firewall.

Privacy settings 隐私设置 control what other people and companies can see: who can view a profile, whether location is shared, whether the history is used for advertising. **Secure sockets layer (SSL) and TLS** encrypt the connection to a website, shown by **https** and a padlock.

Worked example. A school worries about students reaching unsuitable websites and about attacks from outside. Which measure answers which worry? The proxy server filters requests and blocks the unsuitable sites by address, and caches pages that many students open, which also saves bandwidth. The firewall answers the attacks from outside, by examining incoming traffic against its criteria and blocking what does not match, while logging what it sees.

2 Practice

2.1 State one advantage of a high-level language.

[1]

2.2 State two tools found in an IDE. [2]

2.3 Identify two tasks performed by a firewall. [2]

2.4 Describe how a proxy server hides a user's identity from a website. [2]

2.5 Give one threat a firewall cannot protect against, and say why. [2]

3 Exam-style questions

3.1 A compiler translates: [1]

A	B
C	D

- A** one line at a time
- B** the whole program at once
- C** only the errors
- D** nothing

3.2 A programmer is writing and testing a new program.

(a) State whether a compiler or an interpreter is more useful while testing, and why. [2]

(b) State two differences between a compiler and an interpreter. [2]

3.3 A company protects its network and its employees' privacy.

(a) Describe three tasks performed by a firewall. [3]

(b) Describe two tasks performed by a proxy server that a firewall does not do. [2]

(c) Explain why a firewall alone cannot keep the network safe, giving one example of a threat it cannot stop. [2]

(d) Employees use a website that shows **https** and a padlock. Explain what this tells them about the connection. [2]

(e) Give one privacy setting an employee should check on a social-media account, and explain the risk if it is left open. [2]
