Ethics and Ownership

A-Level Computer Science

Ethics for computing professionals

A **computing professional** is someone whose work — software, systems, networks, data — affects other people. Because the work is technical, others often cannot judge whether it was done well or honestly. So the profession follows shared **ethics** 伦理 (principles for good behaviour).

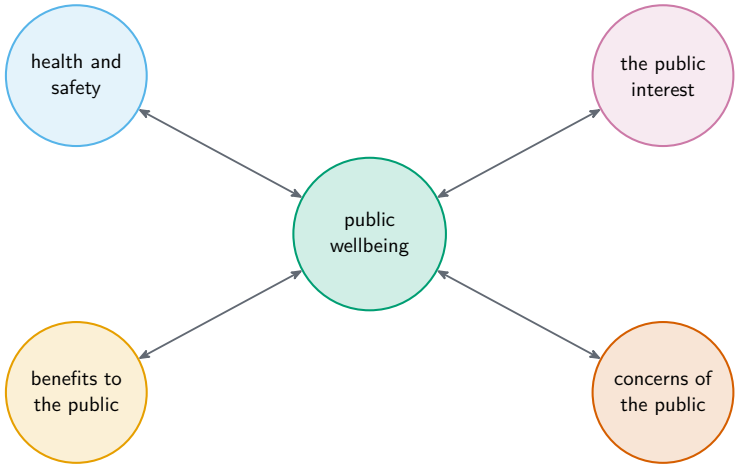
Why ethics matters

- **trust** — users and employers trust professionals to act in their interest. Without that trust, software loses credibility.
- **impact** — software runs medical devices, banking, vehicles. Careless or dishonest work can hurt people.

Professional bodies (BCS, ACM, IEEE) publish codes of ethics for members.

CCTV raises privacy concerns — one of the ethical issues a computing professional must weigh

Discarded electronics (e-waste) are a growing environmental cost of computing



Software development affects the public's wellbeing in several ways

Typical principles

- **public interest first** — protect the safety and welfare of those affected.
- **honesty and competence** — be honest about your skills; don't claim expertise you lack.
- **confidentiality** 保密性 — protect clients' and employers' private information.
- avoid **conflicts of interest** 利益冲突 — don't take work where your interest clashes with the client's.
- keep your skills current; respect **intellectual property** 知识产权 and **privacy** 隐私; treat colleagues fairly.

Joining a professional body

The syllabus names two: the **BCS** (British Computer Society) and the **IEEE** (Institute of Electrical and Electronics Engineers). Both publish a **code of conduct** 行为准则 that members agree to follow. The benefits of joining, in the scheme's words: a set of ethical guidelines to follow, so decisions are not left to personal judgement; training, conferences and publications that keep the member up to date; advice and support, including legal help, when a problem arises; and recognised professional status, so employers and clients trust the member's work. The consequences of **not** joining: no guidance on ethical decisions, so the programmer may act unethically without realising; less credibility with employers and customers, so it is harder to win work; no support in a dispute; and being out of date with developments and law. The purpose of a code of conduct (two marks): to create a safe, respectful and professional working

environment, and to make sure every employee understands what is expected and the consequences of their actions.

Worked example. Explain why a programmer needs to act ethically towards colleagues and towards the public.

Colleagues: treat them fairly and without discrimination; respect their work, their ideas and their confidential information; be honest about mistakes and give credit where it is due; support their development rather than undermine it. The public: protect their personal data and privacy; produce software that is safe, reliable and properly tested, because faults can cause harm; be honest about what the software can do; take on only work within your competence; obey the law and consider the wider effects on society and the environment. Each side earns marks for a reason and its consequence, not for the word "fair" alone.

Acting ethically vs unethically

Acting **ethically** protects users, strengthens reputation, reduces legal risk, and builds trust. Acting **unethically** (skipping testing, hiding bugs, misusing data) can harm real users, lead to dismissal or legal action, damage reputation, and erode trust in technology generally.

When you face a borderline decision: identify whose interests are affected, check the code of ethics and the law, weigh the consequences, ask a trusted senior, and choose the option that protects users above short-term convenience.

Worked example. Your team's new AI hiring tool sorts CVs ten times faster, but you notice it rejects more older applicants. Shipping it pleases your manager, but it treats one group unfairly. The ethical choice is to **hold it back until the bias is fixed** — public interest and fairness come before short-term convenience.

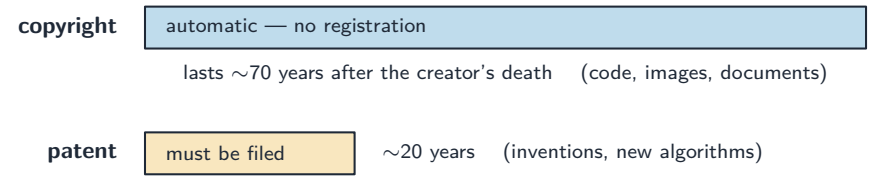
Ethics also applies to users. A student who connects a personal computer to the school network should respect other people's privacy and data, not use social media inappropriately or bully others, not download or share copyrighted material, not introduce malware or try to access systems they are not allowed to, and use the network for the purpose it was provided. A "give three ethical considerations" answer lists three of these.

Copyright

Copyright 版权 is the legal right of the creator of an original work to control how it is **copied, distributed, modified and performed**. It applies automatically (no registration) to source code, software, documents, images, audio and video.

Without copyright, anyone could copy software freely, the developer would not be paid, and plagiarism would be legal. With copyright, developers can earn from their work (encouraging more software), users know who made it, and re-use happens on the developer's terms through licensing. Copyright lasts a long time (often 70 years after the creator's death). General ideas and algorithms are not covered by copyright but may be covered by a **patent** 专利.

Why a programmer should copyright a program, in the scheme's words: to be identified as the owner and author (formal recognition of ownership); so that there are legal consequences if anyone copies or steals it; to restrict competitors from selling the same work; and to be able to earn money by licensing it. Copyright applies to the program as written; a different program that does the same job does not infringe it.



Copyright is automatic and long-lasting; a patent must be filed and lasts about 20 years

Software licences

A **software licence** 软件许可证 is a contract granting permission to use software on the owner's terms; choosing and applying one is called **software licencing**.

Commercial (proprietary)

- **commercial software** is sold: you **buy a licence**; the software is used only within its terms.
- the source code is not given (a **proprietary** 专有 product); you cannot modify or redistribute it.
- examples: Microsoft Office, Adobe Photoshop, most games.

Used when the developer wants **revenue** per user and to **keep control** of the code.

Open-source

- the source code is **public**; users can read, modify and redistribute it (**open-source** 开源).

- **permissive** licences (MIT, BSD) allow almost any use; **copyleft** 著佐权 licences (GPL) require that modified versions are released under the same licence ("share-alike").
- the **Free Software Foundation** (FSF) and the **Open Source Initiative** (OSI) promote and approve open-source licences.

The syllabus names both, and they are marked as distinct answers. **Free Software** (the FSF's term) means free as in freedom, not price: the user may run the program for any purpose, study and change it (so the source code must be available), redistribute copies, and distribute modified versions; a fee may still be charged for a copy. **Open Source** (the OSI's definition) requires that the source code is available, that the program may be modified and redistributed, and that the licence does not discriminate against any person or field of use. "Identify two licence types that let other people edit and redistribute the program" is answered with these two.

- examples: Linux, Python, Apache.

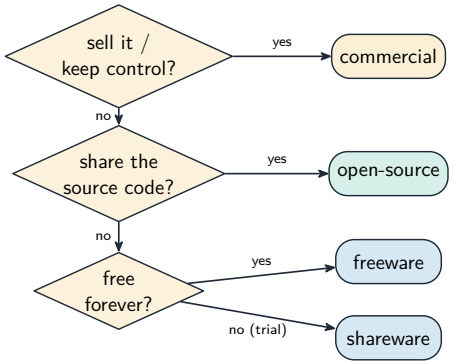
Used when the developer wants the software **widely used and improved by the community**.

Freeware and shareware

- **freeware** 免费软件 — free of charge, no source code, may be redistributed but not modified (Acrobat Reader, WhatsApp).
- **shareware** 共享软件 — free for a trial period, then you pay to keep using it; no source code.

The scheme's descriptions: **shareware** is distributed free for a trial (a limited time or limited features) and the user pays to continue using the full version; **commercial software** is sold for a fee, the source code is not supplied, the licence protects the developer's intellectual property, and the fee usually buys support and updates. Benefits of shareware to the programmer: users can try the program before buying, so they are more likely to purchase; it spreads widely at almost no advertising cost; and those who keep it pay. Benefits of a commercial licence: the developer earns a fee for every copy; the code and its rights stay protected; and the income funds support, updates and further development.

Type	Cost	Source	Redistribute	Modify
Commercial	Paid	No	No	No
Open-source	Free	Yes	Yes	Often, with conditions
Freeware	Free	No	Yes	No
Shareware	Free trial, then paid	No	Sometimes	No



Choosing a licence from the developer's goal

To justify a licence choice, link it to the developer's goal (revenue, reach, community), the user's needs (cost, customising), and the use case.

Worked example. A programmer has written a game to sell to the public. Identify the most appropriate licence and justify it.

A commercial licence: the game is sold for a fee, so the programmer earns money from every copy; the source code is not released, so nobody can copy the game or change it and sell it as their own; the licence protects the intellectual property; and buyers receive updates and support. Open source would not fit, because the source code would be available, so the game could be copied, changed and redistributed without payment.

Worked example. A program helps shoppers by reading product labels aloud. Explain why an open source licence might not be appropriate.

The source code would be accessible, so it could be changed; a changed version might output the wrong product information, so shoppers could buy the wrong item; and the programmer would lose control over the quality and safety of what is distributed under the program's name. Going the other way, programs are released as open source so that other developers can improve and extend them, so that they are adopted widely at no cost, and so that users can adapt them to their own needs.

Artificial Intelligence (AI)

Artificial intelligence 人工智能 builds systems that do tasks once thought to need **human intelligence** — recognising speech and images, translating, playing games, driving.

Most modern AI uses **machine learning** 机器学习 — algorithms that improve at a

task by learning patterns from large amounts of data, instead of being programmed step by step. **Deep learning** 深度学习, using **neural networks** 神经网络 with many layers, is the leading approach today.

Everyday examples

AI tasks split into two kinds — **understanding input**, and **producing output or decisions**.

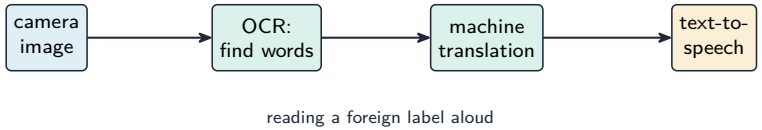
Understanding input:

- **speech recognition** 语音识别 — spoken words to text (voice assistants).
- **image recognition** 图像识别 — finding objects, faces or text in images.

Producing output or decisions:

- **machine translation** 机器翻译 — automatic translation between languages.
- **recommendation systems** 推荐系统 — suggesting products, videos or music.
- **autonomous vehicles** 自动驾驶汽车 and robots.

A common exam scenario: a program reads a label with a camera, translates it, and reads it aloud — using **optical character recognition** 光学字符识别 to find the words, machine translation to convert them, and **text-to-speech** 文本转语音 for the audio.



A common scenario: OCR → machine translation → text-to-speech reads a foreign label aloud

A four-mark "explain how AI is used" answer follows the pipeline step by step: image recognition (OCR) analyses the pixels of the photograph to locate the characters; the patterns of pixels are converted into individual characters and words; machine translation converts the words into the user's language; and text-to-speech produces the spoken output. Each step is a mark.

Benefits

- **accessibility** — speech/image AI helps users with impairments; translation helps non-native speakers.
- **productivity** — automating repetitive tasks frees people for creative work.

- **decision support** — AI spots patterns in huge datasets (medical diagnosis, fraud detection).
- **always available**, and **personalised** to each user.

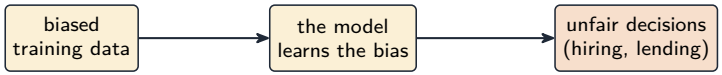
Impacts: social, economic, environmental

The syllabus asks for the impact of AI under three headings, and a question names one of them. Give the impact **and** its consequence.

- **Social**: benefits — a label-reading program helps people with a visual impairment, people who cannot read the language, and people with reading difficulties; facial recognition at an airport speeds up identity checks and can stop wanted people entering. Harms — facial recognition can misidentify people and tracks everyone without consent, so privacy is lost; students who use AI to do their homework may not develop reasoning and problem-solving skills, may rely on it instead of learning, and may lose the collaboration and face-to-face communication that working together brings.
- **Economic**: an AI fault-diagnosis module in a repair garage diagnoses faults faster and more accurately, so more vehicles are repaired per day and costs fall; but fewer skilled mechanics may be needed, so jobs are lost, and the module must be bought and maintained. More generally, AI raises productivity and creates new jobs in some fields while removing routine jobs in others.
- **Environmental**: training and running large models uses a great deal of electricity and water for cooling in data centres, and the hardware becomes e-waste; on the other side, AI is used to cut energy use in buildings, optimise transport and monitor the environment.
- **Ethical** (the classroom question): an AI that marks work or watches students must be fair to every student, must not leak their data, must be explainable when it makes a decision about them, and must not replace the judgement of a teacher where that matters.

Concerns

- **bias** 偏见 — unfair patterns in the training data become unfair AI decisions (hiring, lending).
- **job displacement** — AI may replace some roles.
- **privacy** — training often uses large amounts of personal data.
- **transparency** — large models are "black boxes", hard to explain.
- **accountability** — when AI is wrong, who is responsible: developer, user, or operator?
- **misuse** — deepfakes, misinformation, surveillance.



How bias gets into AI: biased data → a biased model → unfair decisions

Professionals must **understand the limits** of the AI they build, **inform users**, and **reduce harm**.

Definitions the examiner accepts

A definition question is marked against fixed wording. Learn these exactly.

Term	Definition
ethics	the moral principles that govern how a person behaves, in a profession the standards set out in its code of conduct
code of conduct	the rules an organisation or professional body sets for how its members must behave
copyright	the legal right of the creator of an original work to control how it is copied, distributed and modified
software licence	the legal agreement that states how a piece of software may be used, copied and distributed
commercial software	software sold for a fee, without source code, under a licence that protects the developer's intellectual property
free software (FSF)	software whose users are free to run, study, change and redistribute it, so its source code is available
open source (OSI)	software whose source code is available and which may be modified and redistributed under its licence
shareware	software distributed free for a trial period or with limited features, after which the user pays to continue
freeware	software that is free of charge to use and copy, but whose source code is not released and may not be modified
artificial intelligence	computer systems that perform tasks which normally need human intelligence, such as recognising images and speech
machine learning	a form of AI in which a system improves at a task by learning patterns from data rather than by explicit programming

Exam tips

- Answer ethics questions against a **professional code of conduct** (public interest, competence, honesty), not personal opinion.
- Distinguish **copyright** (protects the expression) from a **patent** (protects an invention).
- Compare software licences: proprietary, open-source, freeware, shareware and FOSS.

Common mistakes

- Giving a personal opinion ("it is wrong") instead of a reason with a consequence ("faulty software could harm users, so it must be tested").
- Treating free software and freeware as the same thing. Free software is about the freedom to study and change the code; freeware is merely free of charge.
- Saying open source means free of charge. It means the source code is available and may be modified and redistributed; a fee may still be charged.
- Writing that copyright must be registered. It applies automatically to the work as written.
- Naming an impact without its consequence. "Job losses" scores when it is tied to why: the AI does the diagnosis, so fewer mechanics are needed.
- Describing what AI is instead of how it is used. The marks are for the steps: recognise, convert, translate, speak.