Software

IGCSE Computer Science · Vocabulary in context

Word	Example
Hardware /ˈhɑːdweə/	**Hardware** is the parts of a computer you can touch.
physical /ˈfɪzɪkl/	A keyboard is a **physical** part of the computer.
Software /ˈsɒftweə/	**Software** is the programs that run on a computer.
programs /ˈprəʊgræmz/	The computer runs many **programs** at once.
System software /ˈsɪstəm ˈsɒftweə/	**System software** runs the computer itself.
operating system /ˈɒpəreɪtɪŋ ˈsɪstəm/	Windows is an **operating system**. Windows
compiler /kəmˈpaɪlə/	The **compiler** turns the whole program into machine code.
linker /ˈlɪŋkə/	The **linker** joins compiled files into one program.
device driver /dɪˈvaɪs ˈdraɪvə/	A **device driver** lets the computer use a printer.
utility software /juːˈtɪlɪti ˈsɒftweə/	**Utility software** looks after the computer, such as by finding viruses.
firmware /ˈfɜːmweə/	The **firmware** in a printer tells its parts what to do.
bootloader /ˈbuːtləʊdə/	The **bootloader** starts the operating system when you turn the computer on.
Application software /ˌæplɪˈkeɪʃn ˈsɒftweə/	**Application software** helps users do tasks, like writing a letter.

exercises.lol exercises.com.cn

Word	Example
word processing /wɜːd ˈprəʊsesɪŋ/	She uses **word processing** software to write essays.
spreadsheet /ˈspredʃiːt/	A **spreadsheet** does calculations in rows and columns.
database management system /ˈdeɪtəbeɪs ˈmænɪdʒmənt ˈsɪstəm/	A **database management system** stores and controls the data.
web browser /web ˈbraʊzə/	Use a **web browser** to visit websites.
providing a user interface /prəˈvaɪdɪŋ ə ˈjuːzə ˈɪntəfeɪs/	The operating system is **providing a user interface** so you can control the computer.
user interface /ˈjuːzə ˈɪntəfeɪs/	The **user interface** is the screen you use to control the program.
managing peripherals /ˈmænɪdʒɪŋ pəˈrɪfərəlz/	**Managing peripherals** means controlling devices like printers.
peripherals /pəˈrɪfərəlz/	Keyboards and printers are **peripherals**.
managing multitasking /ˈmænɪdʒɪŋ ˈmʌltitæskɪŋ/	**Managing multitasking** lets several programs run at once.
multitasking /ˈmʌltitæskɪŋ/	**Multitasking** lets you play music while you type.
providing a platform /prəˈvaɪdɪŋ ə ˈplætfɔːm/	The operating system is **providing a platform** for apps to run on.
platform /ˈplætfɔːm/	Windows is a **platform** for running apps. Windows
interrupt /ˈɪntərʌpt/	Pressing a key sends an **interrupt** to the processor.
processor /ˈprəʊsesə/	The **processor** runs the program's instructions.

Word	Example
priorities	The operating system gives some tasks higher **priorities**.
central processing unit /'sentrəl 'prəʊsesɪŋ 'juːnɪt/	The **central processing unit** carries out program instructions.
programming language /'prəʊgræmɪŋ 'læŋgwɪdʒ/	Python is a **programming language**. Python
high-level language /haɪ 'levl 'læŋgwɪdʒ/	A **high-level language** is close to English.
low-level language /ləʊ 'levl 'læŋgwɪdʒ/	A **low-level language** is close to what the CPU understands. CPU
machine code /mə'ʃiːn kəʊd/	The processor runs **machine code**: only 0s and 1s. 0 1
assembly language /ə'semblɪ 'læŋgwɪdʒ/	**Assembly language** uses short words such as ADD and LDM. ADD LDM
specific /spə'sɪfɪk/	Assembly code is **specific** to one type of CPU. CPU
translator /træn'sleɪtə/	A **translator** turns source code into machine code.
interpreter /ɪn'tɜːprɪtə/	An **interpreter** runs the program one line at a time.
integrated development environment /'ɪntɪgreɪtɪd dɪ'veləpmənt en'vaɪrənmənt/	Python's IDLE is an **integrated development environment**. Python IDLE
code editor /kəʊd 'edɪtə/	You type your program in a **code editor**.
error diagnostics /'erə ˌdaɪəg'nɒstɪks/	**Error diagnostics** show where the mistakes are.

exercises.lol exercises.com.cn

Word	Example
run time environment /rʌn taɪm en'vaɪrənmənt/	The **run time environment** lets you run and test the program.
assembler /ə'semblə/	The **assembler** turns assembly language into machine code.
auto-completion /'ɔːtəʊ kəm'pliːʃn/	**Auto-completion** suggests the rest of a word as you type.
auto-correction /'ɔːtəʊ kə'rekʃn/	**Auto-correction** fixes simple typing mistakes.
prettyprinting /'pretɪprɪntɪŋ/	**Prettyprinting** colours keywords to make code easier to read.