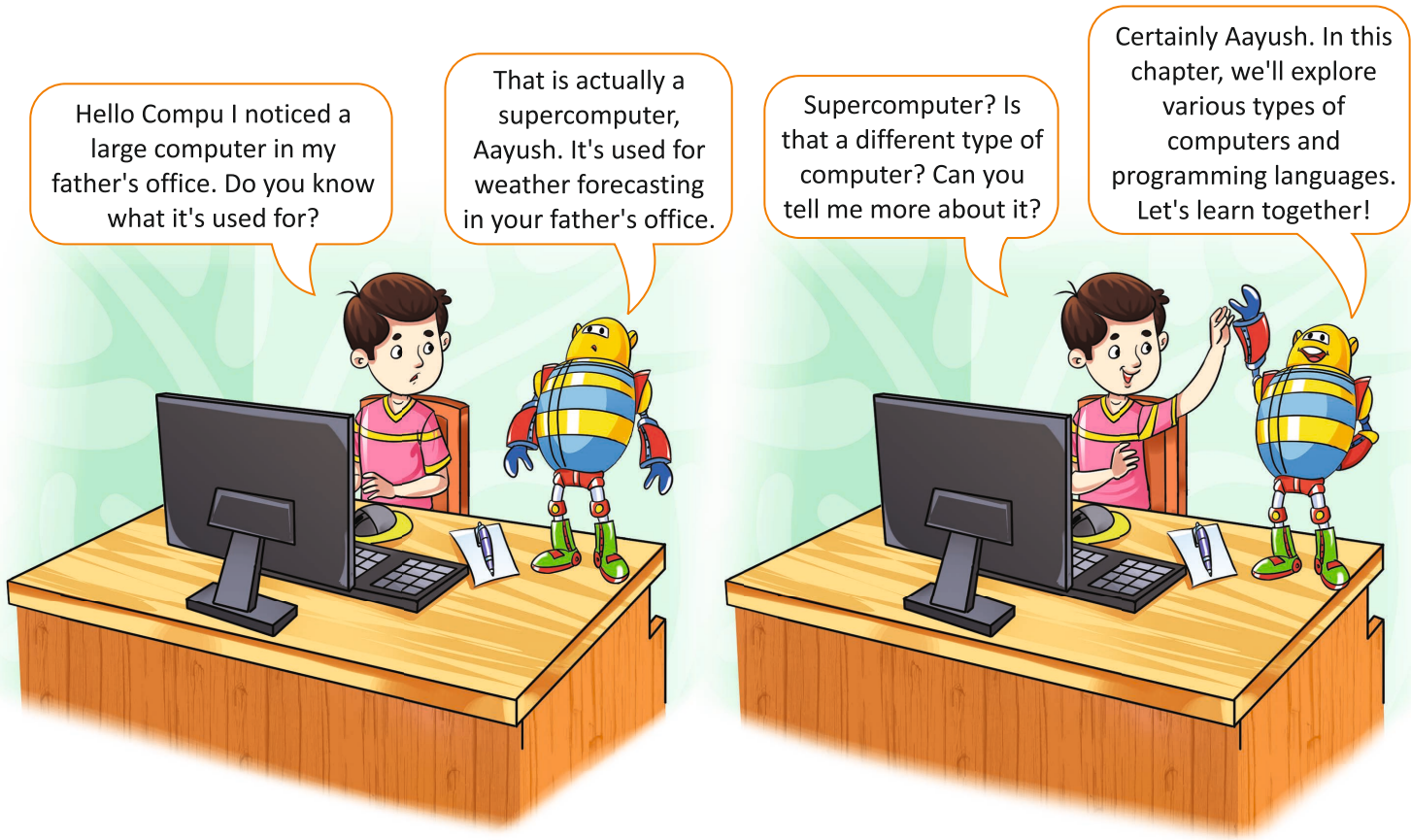


Computer Categories and Languages

KEY POINTS

► Types of Computers

► Computer Languages



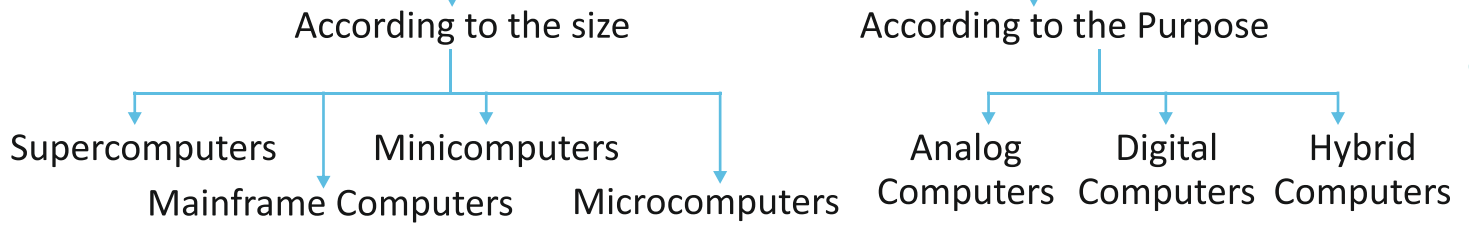
A computer is a machine that has been programmed to read and understand symbols. It can quickly store and retrieve data. Computers can easily perform complex tasks and respond to requests quickly. Modern computers are digital and electronic in nature. Let us learn about different types of computers.

TYPES OF COMPUTERS

Computers can be classified into two main categories by their size and purpose.



TYPES OF COMPUTERS



Types of Computers on the Basis of Size

On the basis of size, computers can be categorised as follows:

Supercomputers

Supercomputers are the biggest and fastest computers in terms of speed of processing data. These are the most expensive computers. These are designed such that they can process a huge amount of data, like processing trillions of instructions or data just in a second. It is because of the thousands of interconnected processors in these computers. These



Fig. 1.1—Supercomputer

computers are used in scientific and engineering applications such as weather forecasting, stock analysis, scientific simulations, space exploration, nuclear energy research, and much more. Some examples of supercomputers are INTEL ASIC RED, PARAM-1000, IBM Roadburner, and CRAY-XMP-14.



Brainy Fact

Supercomputer was first developed by Roger Cray in 1976.

Mainframe computers

Like supercomputers, mainframe computers support hundreds of users at the same time and are expensive. They also support multiple programs, simultaneously. Hence, they can execute different processes, simultaneously. They have high storage capacity and great performance. All these features make the mainframe computer ideal for big organisations like banking, telecom sectors, trading companies, health care, retail sector, etc., which process a high volume of data in general. They run smoothly for a long time and hence have a long life. Some examples of mainframe computers are IBM's Z135 and Hitachi's z900.



Fig. 1.2—Mainframe computer



Minicomputers

Minicomputers are medium size multiprocessing computers. They have two or more processors and support 4 to 200 users at one time. They are lightweight computers and because of their weight, they can be carried anywhere. They are used in places like institutes or departments for different work like billing, accounting, inventory management, etc. They are smaller than mainframe computers but larger in comparison to the microcomputers. Some examples of minicomputers are Kangaroo MD2B, Apple's Mac Mini, and CDC 160A.



Fig. 1.3—Minicomputer

Microcomputers

Microcomputers are general-purpose computers, designed for individual use. They are also called **personal computers**. They consist of microprocessors as central processing units (CPU), memory, input unit, and output unit. These types of computers are suitable for personal work such as making an assignment, watching movie, or for office work, etc.

Laptops, desktop computers, palmtops, tablets, etc., are some different types of microcomputers.



Fig. 1.4—Laptop



Arrange the pictures of different types of computers from old magazines and paste them into your computer notebook.

Types of Computers on the Basis of Purpose

On the basis of purpose, computers can be categorised as follows:

Analog Computer

An analog computer is particularly designed to process analogue data. The data that changes continuously and cannot have discrete values is called **analogue data**. So, an analog computer is used where we do not need exact values or need approximate values such as speed,



Fig. 1.5—Analog weighing machine

temperature, pressure, etc. It directly accepts the data from the measuring device without first converting it into numbers. It measures the continuous changes in physical quantity. It gives output as a reading on a dial or scale. For example, speedometer, mercury thermometer, voltmeter, etc.

Digital Computer

Digital computers can easily perform logical calculations at high speed. It takes raw data as input and processes it with programs stored in its memory to give the final output. It gives more accurate and faster results. It only understands the binary input 0 and 1, so the raw input data is converted to 0 and 1 by the computer and then it is processed by the computer to produce the result. All modern computers, like laptops, desktops, tablets, smartphones are digital computers.



Fig. 1.6—Digital weighing machine

Hybrid Computer

We know that hybrid means made by combining two different things. The hybrid computer is a combination of both analog and digital computers. They are fast like an analog computer and have memory and accuracy like a digital computer. So, it is widely used in specialised applications where both analog and digital data are required to be processed. A processor which is used in petrol pumps that converts the measurement of fuel flow into quantity and price is an example of a hybrid computer.



Fig. 1.7—Petrol pump machine



Answer This

Match the following.

1. This is the biggest and fastest computer.
 2. This computer can support 4 to 200 users at a time.
 3. Hitachi's z900 is an example of this computer.
 4. Desktop computer is an example of this computer.
 5. Thermometer is an example of this computer.
- a. Minicomputer
 - b. Microcomputer
 - c. Supercomputer
 - d. Digital computer
 - e. Mainframe Computer



COMPUTER LANGUAGES

Computer language is a coded syntax used to communicate with a computer system. This is the only language that computers understand. This is also known as a **programming language**. Computer language helps in establishing a flow of communication between different software programs, enabling a user to instruct what commands the computer should perform to process data. Mentioned here are some of the most popular computer programming languages that are used today:

- ❖ Java
- ❖ Javascript
- ❖ SQL
- ❖ Swift
- ❖ C++
- ❖ PHP (Hypertext Preprocessor)
- ❖ Python
- ❖ Pearl
- ❖ C



Brainy Fact

Lady Ada Lovelace is considered as the first computer programmer.



Low-level Languages

A low-level programming language is a type of programming language that gives instructions to computers in the form of machine language which is difficult to understand by humans. Consequently, the commands or functions in this kind of language map closely to all the processor instructions, which generally refer to either the machine code or assembly language.

In other words, low-level language is designed to operate and handle the entire hardware and instruction set of a computer, directly. Machine code, for instance, is a low-level language because it directly runs on the processor.

The features of low-level programming language are mentioned as follows:

- ❖ It is a type of direct memory management.
- ❖ The language has little to no abstraction from the hardware.
- ❖ The language offers registered access, that is, it makes direct use of the internal registers.
- ❖ It gives an amazing performance.

Two main examples of low-level languages are:

- ❖ Machine language
- ❖ Assembly language





Machine Language (First Generation Language)

Machine language is the elemental language of computers. It is read by the computer's central processing unit (CPU) and is composed of digital binary numbers. It looks like a very long sequence of 0s and 1s. Ultimately, the source code of every human-readable programming language must be translated into machine language by a compiler or an interpreter, because binary code is the only language that computer hardware can understand.

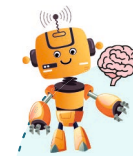
Each CPU has its specific machine language. The processor reads and handles instructions, which tell the CPU to perform a simple task. Instructions are comprised of a certain number of bits. For example, if instructions for a particular processor are 8 bits, then the first 4 bits part tells the computer what to do and the second 4 bits tell the computer what data to use.

Assembly Language (Second Generation Language)

An assembly language is a low-level programming language that is used for microprocessors and other programmable devices. It is not just a single language but a group of various languages. An assembly language makes use of a symbolic representation of the machine code, which is needed to program a certain kind of CPU architecture.

Generally, the assembly languages consist of the following:

- ❖ It has comments, just as any other programming language.
- ❖ It uses directives, which are used to control how the program is assembled, and allocates static variables among other things.
- ❖ It uses labels to associate a name with an address.
- ❖ It has instructions, which are nothing but an architecture-specific string that specifies what type of instruction should be generated.
- ❖ It incorporates operands that are used to select a specific version of the instruction.



Brainy Fact

An assembly language is often abbreviated as **asm**.

Some of the features of assembly language are as follows:

- ❖ It is a machine-dependent language, which means that they are architecture dependent as well as model-specific in nature.
- ❖ They usually are low-level languages.
- ❖ Almost all operating system codes contain a type of assembly for a crucial part, for example, boot code and context switch.



High-level Languages (Third Generation Language)

A high-level language, commonly abbreviated as HLL, is a type of programming language that allows a programmer to write programs using English words and symbols that are more or less independent of a particular type of computer. These types of languages are considered high-level because they are closer to human languages. A few examples of HLL are stated as follows:

- ❖ COBOL
- ❖ FORTRAN
- ❖ Pascal
- ❖ BASIC
- ❖ C
- ❖ Python



Fig. 1.8—Fortran Python Programming Language

Some of the features of high-level computer languages are as follows:

- ❖ HLLs are very close to human languages. The instructions written in HLL are very similar to English statements, thus making it very easy to learn.
- ❖ The logic used to write HLL is very easy and mostly written in English. So, any errors can easily be detected in HLL.
- ❖ The program written in HLL is not dependent on the machine. This means a program written on one computer can easily be run on another system that has a different architecture.

High-level computer languages do not use mnemonic codes. Each instruction in the high-level language is translated into the machine language instructions and then translated back to the high-level language with the result, using a translator.

Translator and Its Types

When a program is written in a high-level language, it is called **source code**. To convert this source code into machine code, we need translators. In other words, a translator takes a program that is written in the source language as input and then converts it into a program in the said language as the output. Furthermore, a translator also detects and reports any errors during this whole translation process.





The different types of translators are as follows:

Compiler

A compiler is a type of translator that is used to convert programs written in high-level languages into low-level languages. Compilers translate the entire program at once and can also report all the errors encountered in the source program during the whole translation process.

Interpreter

An interpreter is another kind of translator used to convert programs from high-level languages into low-level languages. The interpreter translates the program line-by-line and reports all the errors encountered during the translation process itself. The interpreter will directly execute the specified operation in the source program, the moment an input is given by the user. It gives us a better error diagnostics than that of a compiler.

The differences between compiler and interpreter are as follows:

Table 1.1 – Difference between Compiler and Interpreter

Compiler	Interpreter
It performs the translation of a program as a whole.	It performs line-by-line translation.
It is faster.	It is slower.
It requires more memory.	Memory usage is comparatively efficient.
It is used in programming languages such as C and C++.	It is used in programming languages such as Python, BASIC, and Ruby.

Assembler

The assembler is a type of translator that is used to translate the assembly language code into the machine language code. Assembly language consists of mnemonics or pseudo code which is converted into binary language by the assembler.

Fourth Generation Language

In the simplest of terms, a fourth-generation programming language (also known as 4GL) is a group of programming languages that attempts to get closer than a third-generation programming language to human language, a form of thinking and ideation.





Some of the features of the fourth-generation programming language are as follows:

- ❖ 4GLs are designed in such a manner that they reduce the effort, overall time and cost of the entire software development.
- ❖ 4GLs are more programmer-friendly and thus, widen the programming efficiency by making use of English-like words and phrases. Whenever needed, 4GLs also make use of graphical interfaces, icons, and symbolical representations.
- ❖ Many 4GLs are associated with data processing and databases that allow the efficient development of business-oriented systems.

Fifth Generation Language

After a lot of development in computer programming languages, the current programming language generation is fifth. This language is based on solving problems using constraints given to the program, rather than using an algorithm written by a programmer. Most constraint-based and logic programming languages and some declarative languages are fifth generation languages.

Fourth-generation programming languages are designed to build specific programs whereas fifth-generation languages are designed to make the computer solve the problem for you. This way, the programmer only needs to worry about what problems need to be solved and what conditions need to be met, without worrying about how to implement a routine or algorithm to solve them. These languages are mainly developed for fields like Artificial Neural Networks and Artificial Intelligence. Examples of fifth generation programming languages are Mercury, Prolog, and OPS5.

Word Glossary

- **Programming language** : A coded syntax is used to communicate with a computer system.
- **Hybrid Computer** : A combination of both analog and digital computers.
- **Machine language** : It is the lowest and the most elementary level of programming language which a computer understands.
- **Assembly language** : A low-level programming language that is used for microprocessors and other programmable devices.
- **Mnemonic codes** : An abbreviated term that is used to define a specific command or function.
- **Compiler** : A type of translator that is used to convert programs written in high-level languages to low-level languages. It translates the entire program at once.
- **Interpreter** : A type of translator that translates the program line-by-line and reports all the errors encountered during the translation process itself.
- **Assembler** : It is a computer program which translates assembly language to machine language format.



Let's Go Over

- Computers can be classified based on their size and power.
- Computer language is a coded syntax used to communicate with a computer system.
- A low-level programming language is a type of programming language that offers little or no abstraction whatsoever from a computer's instruction set architecture.
- An assembly language is a type of low-level programming language that is used for microprocessors and other programmable devices.
- A high-level language allows a programmer to write programs that are more or less independent of a particular type of computer.
- The translator takes a program that is written in the source language as input and then converts it into a program in the said language as the output.
- The fourth-generation programming language is a group of programming languages that attempts to get closer than third-generation programming languages to human language, a form of thinking and ideation.
- The fifth-generation languages are mainly developed for fields like Artificial Neural Networks and Artificial Intelligence.



A. Tick (✓) the correct answer.

- Which of the following types of computers is a laptop?

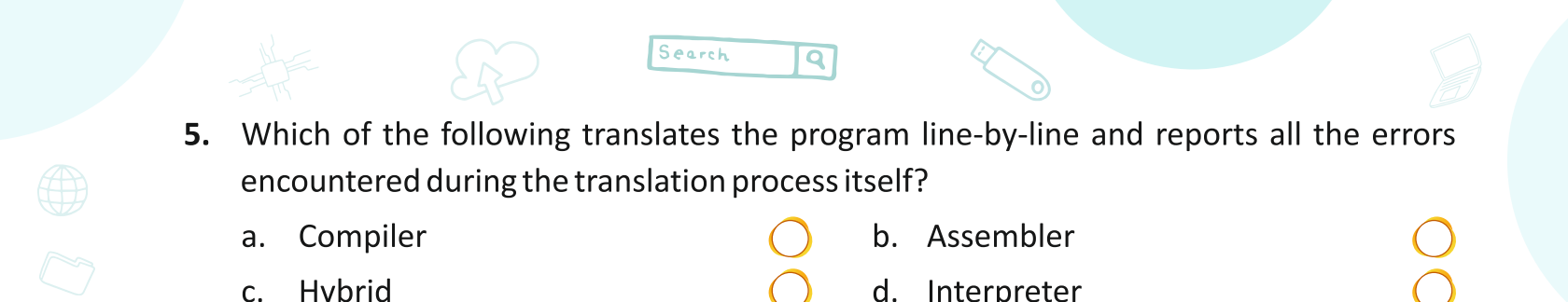
a. Minicomputer	<input type="radio"/>	b. Microcomputer	<input type="radio"/>
c. Supercomputer	<input type="radio"/>	d. Mainframe computer	<input type="radio"/>
- Which of the following types of computers is a petrol pump machine?

a. Analog	<input type="radio"/>	b. Digital	<input type="radio"/>
c. Hybrid	<input type="radio"/>	d. All of these	<input type="radio"/>
- Which of the following is not an example of a programming language?

a. Python	<input type="radio"/>	b. Java	<input type="radio"/>
c. Paint	<input type="radio"/>	d. C++	<input type="radio"/>
- Which generation language is Assembly language?

a. First	<input type="radio"/>	b. Second	<input type="radio"/>
c. Third	<input type="radio"/>	d. Fourth	<input type="radio"/>



- 
5. Which of the following translates the program line-by-line and reports all the errors encountered during the translation process itself?
- a. Compiler b. Assembler
 c. Hybrid d. Interpreter

B. Fill in the blanks.

1. A _____ computer is small in size and is usually only used by one person at a time.
2. An assembly language is often abbreviated as _____.
3. _____ is a high-level programming language.
4. _____ translates the entire program at once and can also report all the errors encountered in the source program during the whole translation process.
5. _____ are designed in such a manner that they reduce the effort, overall time, and cost of the entire software development.






C. State whether true or false.

1. Machine code is a low-level language because it directly runs on the processor. _____
2. An assembly language does not use labels to associate a name with an address. _____
3. A translator cannot detect and report any errors during this whole translation process. _____
4. Interpreter will not give you a better error diagnostic than that of a compiler. _____
5. A mainframe computer is a large and powerful multi-user computer. _____

D. Name the following.

1. A type of computer that is capable of supporting hundreds of thousands of users at the same time. _____
2. A type of programming language that is designed to operate and handle the entire hardware and the instruction set of a computer directly. _____
3. A type of programming language that makes use of a symbolic representation of the machine language. _____
4. A type of translator that is used to convert programs written in high-level languages to low-level languages. _____



- 
- 
- 
- 
- 
5. A type of translator that is used to translate the assembly language code into the machine language code.

E. Answer the following.

1. List the different types of computers.
2. How is Microcomputer different from Supercomputer?
3. Give any two examples of the different types of computer programming languages.
4. List the features of low-level programming language.
5. What is the use of fifth-generation language?

F. Application-based questions.

1. Suresh's father works in the stock exchange. He keeps an eye on the share market. What kind of computer would he use to check all that?
2. Kumar wants to do his project work at home. What type of a computer will he use for this purpose?

Activity 

Research different types of translators mentioned in the chapter and understand how each one is different from the others. Create a PowerPoint presentation on the same.

1. Apply the Angles theme.
2. Use different slide layouts, such as Title Slide, Title and Content, and Picture with Caption.
3. Change the colour scheme of the slide in the presentation.
4. Add appropriate images.

Lab Time 

Create a word document on the topic 'Types of programming languages'. Also, create a flowchart on the same.

Life Skills 

Always keep your computer clean and covered after use.

Connect to Subject 

Prepare a flowchart on different types of computers. Paste this chart in your Art notebook.





Note to the Teacher

- Explain the meaning of programming languages by giving examples.
- Encourage the students to learn about all types of computers, not just the desktops, laptops, and tablets.
- Encourage the students to gather information about all types of computers from the Internet or magazines.
- Ask the students to read more about Lady Ada Lovelace and discuss the importance of programming languages.
- Explain the use of fifth-generation computer.

