Throughout life, we always find something to measure: the amount of food we need to cook for the family, the length and width of that couch you want to put in the room, our weight, and height. The latter is especially exciting: it's really cool to learn that in just one year you grew by a full 2 inches!

Each measurement requires an instrument and its own **unit of measurement**. For example, body weight is measured with scales in kilograms (or pounds), time is measured with clocks in seconds, etc. But how does one measure information?

**Bit: the smallest unit of information**

The information entered into the computer should be specific and unambiguous. For a long time, people have been using codes. The simplest and most convenient of them were digital. Any information from the names of flowers to the days of the week can be presented in the form of numbers. When processed using a traditional computer, data is encoded numerically and stored in files, ultimately represented by the electrical signals that dictate the computer's operational speed.

For the convenience of distinguishing, signals of two levels are used in classical electronic computers. One of them corresponds to the number 1, and the other to 0. Any letter, sound, or image on a computer is represented by a set of numbers. The numbers 1 and 0 are called **binary**. These are the symbols that make up the language understood and used by the computer. Any information on the computer is represented by binary digits: 1, meaning "**there is a signal**" or **"there is a high signal"** and 0, meaning "**no signal**" or **"there is a low signal"**.

The smallest unit of information is the **bit** **(b)**.

Each digit of the machine binary code carries an amount of information equal to one bit. It can take only one of two values: either 1 or 0. It is very inconvenient to measure information in bits because the numbers come out too large.

**Byte: a sequence of eight bits**

Like people do not consider the mass of ships in grams, larger and hence more convenient units were invented for measuring information as well.

The processing of information takes place in the processor. This is a device that can work with several bits at once (8, 16, 32, 64, ...). Most computers nowadays process 8 bits of information simultaneously, so we needed a new unit of measurement which was called a **byte** **(B)**, which means **8 bits**.

Bit marks are easily confused with byte marks. Note that the abbreviations for bit numbers use the lowercase letter **"b",** while bytes are denoted using the capital **"B"**.

It's important not to confuse bit and byte abbreviations. The lowercase 'b' is used for bits, while the uppercase 'B' denotes bytes. This distinction is important, as mixing them up can lead to significant misunderstandings, especially when measuring data transfer rates or storage capacity.

**Large units of information**

Given the advanced capabilities of modern computers, the storage capacity has expanded to accommodate significantly larger units of information, surpassing the traditional byte scale.

The computer industry has historically used the units **kilobyte**, **megabyte**, and **gigabyte** in at least two slightly different measurement systems, which are slightly contradictory to each other.

* The first one is a decimal-based system, which uses bytes in powers of ten: **kilobyte** (10^3 bytes), **megabyte** (10^6 bytes), **gigabyte** (10^9 bytes), and so on. These units are used by the [International System of Units](https://en.wikipedia.org/wiki/International_System_of_Units) (SI).
* The second one is a binary-based system that uses bytes in powers of two: **kibibyte** (2^10 bytes), **mebibyte** (2^20 bytes), **gibibyte** (2^30 bytes), and so on. This system was actively used to describe computer memory.

To reduce confusion, the [International Electrotechnical Commission](https://en.wikipedia.org/wiki/International_Electrotechnical_Commission) (IEC) proposed using prefixes kilo, mega, and giga only for the decimal-based system. For the binary-based system, they introduced new prefixes: **kibi**, **mebi**, and **gibi**. Here, 'bi' stands for binary. So, a kibibyte (KiB) equals 1,024 bytes, distinguishing it from a kilobyte (kB), which equals 1,000 bytes.

Here is a table with commonly used units of information according to modern international standards.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SI metric** | **Symbol** | **Powers of ten** | **IEC metric** | **Symbol** | **Powers of two** |
| Kilobyte | kB | 10^3 B (1000 B) | Kibibyte | KiB | 2^10 B (or 1024 B) |
| Megabyte | MB | 10^6 B (1000 kB) | Mebibyte | MiB | 2^20 B (or 1024 KiB) |
| Gigabyte | GB | 10^9 B (1000 MB) | Gibibyte | GiB | 2^30 B (or 1024 MiB) |
| Terabyte | TB | 10^12 B (1000 GB) | Tebibyte | TiB | 2^40 B (or 1024 GiB) |
| Petabyte | PB | 10^15 B (1000 TB) | Pebibyte | PiB | 2^50 B (or 1024 TiB) |

Of course, not all units of measurement are listed here. We hope this classification will not confuse you.

**Why does this matter?**

This distinction might seem minor, but it becomes important in certain contexts.

For example, when buying a hard drive, you might notice that a drive advertised as 1 TB only shows about 931 GiB when connected to your computer. This is because the manufacturer uses the decimal system (1 TB = 1,000 GB), while your operating system uses the binary system (1 TiB = 1,024 GiB).

In programming, knowing the difference between these units may also prevent bugs. For instance, allocating memory in MiB instead of MB can result in an incorrect amount of memory being assigned, potentially leading to errors or performance issues.

Understanding these units can help prevent misunderstandings.

**Measurement units conversion**

To strengthen your newly obtained knowledge, let's look at the solution of a rather typical problem where you need to convert **1 GiB** to **KiB**. When we convert bigger units into smaller ones, we need to resort to an arithmetic operation called multiplication:

1 GiB = 1KiB \* 1024 \* 1024 = 1048576 KiB

Accordingly, when you need to convert small units into big ones, you use division. Let's try to convert **16384** bits to **KiB**:

16384 bits = (16384 / 8) / 1024 = 2 KiB

If you want to convert **1 GB** to **kB**, you should multiply the number by a thousand twice:

1 GB = 1kB \* 1000 \* 1000 = 1000000 kB

A frequent mistake is to assume all "kB" measurements refer to 1,024 bytes. Some software or hardware manufacturers might use "kB" to mean 1,000 bytes, leading to confusion. It's always best to verify the context to avoid misunderstandings.

Congratulations, now you have studied one of the basic topics of computer science and are ready to reach new dizzying heights of knowledge.

**Conclusion**

To wrap-up:

* The smallest unit of information is a bit.
* One byte consists of 8 bits.
* For convenience in computer science, we use binary-based units of information such as bytes, kibibytes, mebibytes, etc.
* To convert one kibibyte to bytes, we need to multiply 1 by 210210; to convert one mebibyte, we need to multiply 1 by 220220, etc.
* Knowing when to use decimal (kB, MB) and binary (KiB, MiB) units can prevent confusion, especially in storage and memory calculations.

Understanding these fundamental units and their correct usage can make navigating the digital world much easier!