

Unit 2: Data

Unit 2 is about how computers represent the world as data. It covers binary numbers and number bases, how analog information becomes digital through sampling, how compression shrinks files, how metadata and bias shape what data means, and how to read visualizations and think about privacy.

AP Computer Science Principles

Data

About 12 minutes to read

How to use this guide

Read it in order the first time because the topics build on each other. Binary representation is the foundation, number bases and data abstraction extend it, then sampling explains how the real world becomes bits, and compression, metadata, bias, and visualization explain what happens to data after that. Exam questions often describe a scenario and ask you to pick the right concept or spot the flaw in a claim.

After the first read, use the trap boxes and the tables to review the distinctions that exam questions test most often. Do the worked conversions on paper yourself before checking the math. Finish with the practice questions, then complete the recall check on the last page out loud and note any items you cannot explain yet.

What this unit is worth. Data is about 17 to 22 percent of the AP Computer Science Principles exam, one of the two largest units. The binary conversion skills and the vocabulary (lossy vs lossless, correlation vs causation, metadata, bias) show up as direct multiple-choice questions, so this unit rewards careful memorization.

2.1 Binary Representation: Bits and Bytes

A **bit** is a binary digit, a 0 or a 1. It is the smallest unit of data a computer stores. A **byte** is 8 bits. Every piece of digital information, text, images, sound, video, and even the program instructions themselves, is stored as a sequence of bits.

Binary is base 2. Each position in a binary number is worth a power of 2, read from right to left: 1, 2, 4, 8, 16, 32, 64, 128, and so on. Each step left doubles the value, just as each step left in decimal multiplies by 10. To convert a binary number to decimal, add up the place values wherever the bit is 1.

Worked example: convert 101101_2 to decimal. Write the place values above the bits, starting at 1 on the right and doubling leftward:

PLACE VALUE	32	16	8	4	2	1
Bits	1	0	1	1	0	1

Add the place values where the bit is 1: $32 + 8 + 4 + 1 = 45$. Check: $32 + 8 = 40$, plus 4 = 44, plus 1 = 45. So $101101_2 = 45$. Verify any conversion by adding the place values a second time, slowly.

A second conversion to check your method. Convert 110_2 : place values 4, 2, 1 with bits 1, 1, 0. Add where the bit is 1: $4 + 2 = 6$. So $110_2 = 6$. Another: 11111_2 has all five bits set, so $16 + 8 + 4 + 2 + 1 = 31$.

Trap. The rightmost bit is the ones place. A common error is reading binary left to right with decimal habits, treating 101101_2 as if the leftmost 1 means "one hundred thousand." Always start at the right with 1 and double as you move left.

2.2 Converting Decimal to Binary

To convert decimal to binary, use **repeated division by 2**. Divide the number by 2, record the remainder (0 or 1), then divide the quotient by 2 and repeat until the quotient reaches 0. Read the remainders from bottom to top. That order matters: the last remainder is the leftmost bit.

Worked example: convert 45 to binary.

STEP	DIVISION	QUOTIENT	REMAINDER
1	$45 \div 2$	22	1
2	$22 \div 2$	11	0
3	$11 \div 2$	5	1
4	$5 \div 2$	2	1
5	$2 \div 2$	1	0
6	$1 \div 2$	0	1

Read the remainders from bottom to top: 101101_2 . Now verify by converting back: $32 + 8 + 4 + 1 = 45$.

The conversion round-trips, so it is correct. A second check: 13 in binary. $13 \div 2 = 6$ r1, $6 \div 2 = 3$ r0, $3 \div 2 = 1$ r1, $1 \div 2 = 0$ r1, which reads 1101_2 . Verify: $8 + 4 + 1 = 13$.

2.3 How Many Values Bits Can Hold

With **n bits** you can represent **2^n different values**. Each bit doubles the count. One bit gives 2 values (0, 1). Two bits give 4 values (00, 01, 10, 11). A byte, 8 bits, gives $2^8 = 256$ values, which is the unsigned range 0 through 255. The largest value is always $2^n - 1$, since one of the 2^n values is 0.

The all-ones byte confirms this: $11111111_2 = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255$. When a value does not fit in the bits available, that is **overflow**, and the program needs more bits. This is why data types come in sizes: an 8-bit value caps at 255, a 16-bit value at 65,535.

Trap. 2^n counts the values, starting from 0. The largest value is $2^n - 1$, not 2^n . Questions that ask "how many values can 10 bits represent" want $2^{10} = 1024$, while questions that ask for "the largest unsigned number 10 bits can hold" want 1023.

2.4 Number Bases and Hexadecimal

A **number base** (also called the radix) is how many distinct digits a system uses. Decimal is base 10 with digits 0 through 9. Binary is base 2 with digits 0 and 1. **Hexadecimal** is base 16, with digits 0 through 9 followed by A, B, C, D, E, F for the values 10 through 15.

Hexadecimal exists as shorthand for binary. Each hex digit represents exactly 4 bits, so long binary strings become short and readable. Programmers write colors, memory addresses, and error codes in hex for this reason. To convert hex to decimal, use place values that are powers of 16: $0xFF$ means $15 \times 16 + 15 = 255$, which matches 11111111_2 . Similarly, $0x2A = 2 \times 16 + 10 = 42$, and checking in binary, $00101010_2 = 32 + 8 + 2 = 42$. The exam does not require heavy hex arithmetic, but you should know what hex is and why it is used.

2.5 Data Abstraction

Data abstraction is representing real-world information as data while leaving out unnecessary detail. It manages complexity by deciding what to keep. A date stored as three numbers (month, day, year) instead of a sentence is a data abstraction. A color stored as three numbers for red, green, and blue is a data abstraction. A student record that keeps a name, an ID number, and a grade level, while dropping eye color and favorite food, is a data abstraction.

The key idea is that the abstraction keeps what the program needs and hides the rest. Two programs can use different abstractions of the same thing: a mapping app stores a restaurant as coordinates and hours, while a review app stores it as a name and a rating.

Trap. Data abstraction is not compression. Abstraction decides *what* information to represent. Compression reduces how much space the representation takes. Choosing to store a color as three numbers is abstraction; shrinking a photo file is compression.

2.6 Analog vs Digital Data and Sampling

Analog data is continuous: it can take any value in a range, like the smooth wave of a sound or a thermometer reading that slides between numbers. **Digital data** is discrete: it takes values from a fixed, finite set, like the list of numbers a computer stores. Computers can only store digital data, so analog information must be converted.

Sampling is that conversion. The analog signal is measured at regular intervals, and each measurement is called a **sample**. The **sampling rate** is how many samples are taken per second. Audio CDs sample sound 44,100 times per second. Each sample value is also rounded to the nearest value the bits can hold, so more bits per sample means finer detail.

A higher sampling rate captures the signal more faithfully but produces more data. This is a genuine tradeoff, and exam questions like to test it: doubling the sampling rate doubles the amount of data. There is no way to record every instant of a continuous signal, so some detail between samples is always lost.

Trap. Sampling does not capture the whole signal. It captures snapshots at intervals. If a question asks what happens between samples, the answer is that the information is not recorded, and a low sampling rate can miss fast changes in the signal.

2.7 Lossy vs Lossless Compression

Compression reduces the size of a file. **Lossless compression** shrinks the data in a way that lets the original be reconstructed exactly. Nothing is thrown away. **Lossy compression** permanently discards some data to get a much smaller file. The original can never be fully recovered.

TYPE	WHAT HAPPENS	EXAMPLES
Lossless	File gets smaller; every original bit can be restored.	ZIP archives, PNG images
Lossy	Some data is discarded forever; file gets much smaller.	JPEG photos, MP3 audio

Which one is appropriate depends on the data and its purpose. Use lossless when every bit matters: text documents, source code, spreadsheets, legal or medical records, and line diagrams where sharp edges matter. Use lossy when an approximation is fine and small files matter more: photos on a website, streamed music, video calls. Lossy formats are designed so the discarded detail is hard for humans to notice, which is why a JPEG photo looks fine even though data is gone.

Two facts the exam tests directly. First, compressing an already-compressed file again with a lossy format loses more quality each time. Second, lossy is not a worse version of lossless. For its purpose, a smaller JPEG that loads fast is the better engineering choice, and the quality loss is invisible in normal viewing.

Trap. Decompressing a lossy file does not restore the discarded data. Opening a JPEG gives you back an image, but it is the reduced image, not the original. If a question asks whether the original can be recovered after lossy compression, the answer is no.

2.8 Metadata

Metadata is data about data. It describes a piece of data rather than being the content itself. A photo file carries metadata like the date it was taken, the camera settings, and sometimes the GPS location. A document carries metadata like the author, the creation date, and the word count. A music file carries metadata like the artist, the album, and the track length.

Metadata is stored alongside the data and makes large collections manageable. It is what lets you search your photos by date or sort a music library by artist without opening every file. On the exam, questions often ask you to identify which item in a list is metadata: look for the description of the data, not the content. The text of an essay is data; the author name attached to the file is metadata.

Metadata has a privacy side. A photo shared online can carry a location in its metadata that the sender never intended to share. Stripping metadata before publishing is a basic privacy habit.

2.9 Extracting Information from Data

Raw data becomes useful when you **extract information** from it: find **patterns**, summarize, and draw conclusions. Before any of that, data usually needs **cleaning**: fixing typos and errors, removing duplicates, and handling missing values. Conclusions drawn from dirty data are unreliable no matter how good the analysis is.

A **correlation** means two things tend to vary together. **Causation** means one thing actually causes the other. Correlation does not imply causation. Ice cream sales and drowning deaths both rise in summer, but ice cream does not cause drowning; warm weather drives both. That third factor is the kind of explanation exam questions want you to spot when they present a correlational claim dressed up as a causal one.

Trap. A strong correlation is not evidence of causation by itself. When a question says "researchers found that X is associated with Y" and asks what can be concluded, the safe answer is that X and Y are correlated, and more work would be needed to show that one causes the other.

2.10 Data Bias

Data bias means the data used to train or analyze a system does not fairly represent the people or situations it will be applied to, and the result is unfair or wrong outcomes. A facial recognition system trained mostly on faces from one demographic performs worse on everyone else. A hiring tool trained on a company's past hiring decisions reproduces whatever preferences, fair or not, shaped those decisions. A course recommendation model trained only on students already in advanced classes will steer others away from those classes.

Incomplete data causes the same problem from the other direction: missing groups, missing fields, or missing time periods skew the conclusions. Cleaning data helps, and it includes more than fixing typos: it means checking where the data came from, asking who is missing, and removing or flagging records that distort the picture.

Two points the exam tests. First, a biased algorithm is usually a data problem, not a math problem. The procedure can be correct and the outcome still unfair if the training data was biased. Second, more data does not fix biased data. A million records from the same skewed source produce the same skewed conclusion, just with more confidence.

Trap. Watch for answer choices that say a larger data set automatically removes bias. Size and fairness are different properties. What matters is whether the data represents the population the system will serve.

2.11 Visualizing Data

Choosing the right chart is part of communicating honestly with data. A **bar chart** compares categories. A **line graph** shows change over time. A **pie chart** shows parts of a whole, and works only with a few categories. A **scatter plot** shows the relationship between two variables, which is where you would look for a correlation. A **histogram** shows the distribution of a single variable, like how test scores spread across ranges.

Visualizations can also mislead, and the exam tests the common tricks. A **truncated axis** starts the y-axis above zero, which makes small differences look dramatic. **Cherry-picking** shows only the data points that support a claim and hides the rest. Other tricks include stretching one axis relative to the other, using 3D effects that distort proportions, and leaving off labels so the reader cannot check the scale.

MISLEADING TRICK	WHAT IT DOES	HOW TO SPOT IT
Truncated axis	Exaggerates small differences between values.	Check whether the axis starts at zero.
Cherry-picking	Hides data that contradicts the claim.	Ask what time range or group was left out.
Distorted scale or 3D effects	Makes one value look bigger than the numbers justify.	Compare the visual sizes against the actual values.

2.12 Privacy Concerns

Large data sets create privacy risks even when they look harmless. Removing names is not the same as making data anonymous. A data set with no names can still identify people when combined with other information: a zip code plus a birth date plus a gender is often enough to single out one person. This is called **re-identification**, and it is why "we removed the names" is not a complete privacy answer.

Other concerns the exam names: data collected for one purpose being used for another, metadata leaking location or identity, and large-scale collection making breaches more damaging. Protections include removing identifying fields, **aggregating** data into summaries so individuals disappear into groups, and limiting who can access the raw data. None of these is perfect, which is why privacy questions often have "reduces the risk" rather than "eliminates the risk" as the right answer.

Trap. If an answer choice claims a privacy technique completely guarantees anonymity, be suspicious. Aggregation and de-identification reduce risk. The exam rewards the choice that acknowledges the remaining risk over the one that promises certainty.

Confusions That Cost Points

PAIR	HOW TO KEEP THEM STRAIGHT
Binary place values vs decimal place values	Binary doubles right to left: 1, 2, 4, 8. The rightmost bit is the ones place. Never read binary with decimal place-value habits.
2^n values vs largest value $2^n - 1$	n bits give 2^n values because 0 counts as a value. The largest is one less. 8 bits: 256 values, largest 255.
Data abstraction vs compression	Abstraction chooses what to represent. Compression shrinks the representation. Different jobs.
Sampling vs compression	Sampling converts analog to digital by measuring at intervals. Compression shrinks an already-digital file. Sampling happens first.
Lossy vs lossless	Lossless restores the original exactly (ZIP, PNG). Lossy discards data permanently (JPEG, MP3). Match the type to whether every bit matters.
Metadata vs data	Metadata describes the data. The essay text is data; the author and date attached to the file are metadata.
Correlation vs causation	Correlation is two things varying together. Causation is one causing the other. Look for the third variable before accepting a causal claim.
More data vs representative data	A bigger data set does not fix bias. What matters is whether the data represents the population the system will serve.
De-identified vs anonymous	Removing names reduces risk but does not guarantee anonymity. Combined fields can still re-identify individuals.

Practice Questions

Original questions written for this guide in the style of the AP exam. Answers and explanations are on the next page, so complete the questions before checking them.

1. What is the decimal value of the binary number 11011_2 ?

1. 23
2. 27
3. 29
4. 31

2. A photographer saves the same photo as a JPEG and as a PNG. The JPEG file is much smaller, but when both are opened and examined closely, the PNG shows fine detail that the JPEG is missing. Which statement best explains this?

1. The JPEG uses lossy compression, which permanently discards some image data.
2. The PNG uses lossy compression, which permanently discards some image data.
3. The JPEG uses lossless compression, but the file was corrupted during saving.
4. The PNG file is larger because it stores the photo at a higher resolution.

3. An analog audio signal is converted to digital by measuring its amplitude 44,100 times per second. What is this process called?

1. Compression
2. Abstraction
3. Sampling
4. Encoding

4. Which of the following is an example of metadata for a digital photograph?

1. The colors of the pixels that make up the image
2. The file size of the image in megabytes
3. A compressed copy of the image stored as a thumbnail
4. The date the photo was taken and the camera model used

- 5.** A company trains a resume-screening program on ten years of its own past hiring decisions. The company historically hired mostly men for technical roles. Which concern best describes this situation?
1. The training data is biased, so the program may reproduce the company's past hiring pattern.
 2. The training data is too small for the program to find any pattern.
 3. The program will be fair because it follows the same procedure for every resume.
 4. The program cannot be biased because it does not store the applicants' names.
- 6.** Researchers find that cities with more ice cream shops tend to have higher rates of sunburn. Which conclusion is best supported?
1. Ice cream shops cause sunburn, so limiting them would reduce sunburn rates.
 2. Sunburn causes people to buy ice cream, which explains the relationship.
 3. The two are correlated, and a third factor such as warm weather likely drives both.
 4. There is no relationship between the two, so the data must contain an error.
- 7.** A bar chart comparing average test scores for four classes uses a y-axis that starts at 70 instead of 0. A difference of 3 points between two classes looks enormous. What is misleading about this chart?
1. The chart cherry-picks which classes to display.
 2. The chart uses a truncated axis, which exaggerates small differences.
 3. The chart should be a pie chart instead of a bar chart.
 4. The chart is misleading because bar charts cannot show averages.
- 8.** A research group wants to publish a large data set of student survey responses. Which step best protects the students' privacy?
1. Publishing the raw responses with names removed but all other fields intact.
 2. Removing names and also aggregating or removing fields that could identify individuals, such as zip code combined with birth date.
 3. Publishing the raw responses exactly as collected, since survey data is not sensitive.
 4. Replacing each student's name with a number and publishing everything else unchanged.

Answer Key

- 1. B.** Place values for 11011_2 are 16, 8, 4, 2, 1. Adding where the bit is 1: $16 + 8 + 2 + 1 = 27$. A (23) is 10111_2 , the result of dropping the 8s place and adding the 4s place, which is the classic misread of one bit position. C (29) is 11101_2 , a different bit pattern that adds the 4s place instead of the 2s place. D (31) is 11111_2 , the value when all five bits are set, which this number is not.
- 2. A.** JPEG uses lossy compression: it permanently discards image detail to shrink the file, so the fine detail cannot be recovered. B reverses the formats; PNG is the lossless one here, which is why it kept the detail. C is wrong on two counts: JPEG is lossy, not lossless, and nothing in the question suggests corruption. D invents a resolution difference the question never states; both files are the same photo, and the size difference comes from the compression type.
- 3. C.** Measuring an analog signal at regular intervals to convert it to digital is sampling, and 44,100 measurements per second is the sampling rate. A is wrong because compression shrinks an already-digital file; this process creates the digital data in the first place. B is wrong because abstraction is about choosing what information to represent, not about measuring a signal over time. D is vague: encoding is a broad term for representing data in a format, and it does not name this specific process.
- 4. D.** Metadata is data about data. The date and camera model describe the photo file rather than being the image content. A describes the content itself, the actual image data, not metadata. B is a property of the file's storage, not a description of the content; file size tells you nothing about what the photo shows or when it was taken. C is a smaller copy of the image data, which is still data, not data about the data.
- 5. A.** The training data reflects a historically skewed hiring pattern, so a program trained on it will learn and reproduce that pattern. This is data bias: the procedure can run correctly and still produce unfair outcomes. B is wrong because ten years of decisions is a large data set, and size was never the problem. C confuses a consistent procedure with a fair outcome; applying the same rule to everyone does not help when the rule was learned from biased data. D is wrong because bias does not require names; the pattern lives in the other features the program learns from.
- 6. C.** The finding describes a correlation: the two rise together. The best-supported conclusion names the third factor, warm weather, which plausibly drives both ice cream sales and sunburn. A jumps from correlation to causation without evidence. B invents a causal direction that is equally unsupported; nothing in the data says sunburn causes ice cream buying. D overcorrects by denying the relationship entirely; the correlation is real, it just is not causal.
- 7. B.** Starting the y-axis at 70 instead of 0 is a truncated axis, and it visually exaggerates the 3-point difference. A is wrong because cherry-picking means hiding data points or groups, not adjusting the axis; all four classes are shown. C is wrong because the chart type is fine: a bar chart comparing categories is the right choice here. D is false; bar charts show averages routinely.
- 8. B.** Removing names is not enough, since combinations of other fields can re-identify individuals. The best step removes or aggregates those identifying fields too, which directly reduces the re-identification risk. A leaves the identifying combinations in place, so it does not solve the problem it names. C ignores the privacy concern entirely. D only swaps names for numbers, which is the same as removing names; the other identifying fields remain and the risk is unchanged.

When you check your answers, note which distinction each miss came from. Make a flashcard for that distinction and drill it spaced out over the next few days instead of rereading the whole section. If you

missed one of these questions, the same distinction is worth practicing again in Rycal, where the Data deck has flashcards for it and more practice questions use the same kinds of traps.

One-Page Recall Check

Say each answer out loud before you look back, and mark the ones you cannot finish. Anything you cannot say out loud yet belongs in your flashcard deck. In Rycal, add those items to the Data deck and let spaced review bring them back over the next few days.

- Define a bit and a byte, and explain what is stored as sequences of bits.
- Convert 101101_2 to decimal, showing the place-value addition.
- Convert 45 to binary using repeated division by 2, then verify by converting back.
- State how many values n bits can represent, and give the largest unsigned value for 8 bits.
- Explain what hexadecimal is and why programmers use it as shorthand for binary.
- Define data abstraction and give an example that is not from this guide.
- Explain the difference between analog and digital data.
- Define sampling, sample, and sampling rate, and state the tradeoff a higher rate creates.
- Explain the difference between lossy and lossless compression, with an example of each.
- State when lossy compression is the appropriate choice and when lossless is required.
- Define metadata and name three examples for a file type of your choice.
- Explain why data cleaning comes before extracting information from data.
- Explain the difference between correlation and causation with an example.
- Define data bias and explain why a larger data set does not fix it.
- Name the right chart type for comparing categories, showing change over time, and showing the relationship between two variables.
- Explain what a truncated axis does and how to spot one.
- Explain why removing names does not guarantee anonymity in a published data set.

Where to go next. Turn every missed item above into flashcards and drill them spaced out over several days rather than in one sitting. In Rycal, open the Data deck under AP Computer Science Principles. The deck covers the terms in this guide, and its practice questions target the same traps named here. If you have a test date, add it in the Test Planner. You can also start your next review with a Brain Dump, then check what you missed against this guide.

Key terms for this unit

Bit, Byte, Binary number, Base (radix), Decimal, Hexadecimal, Data abstraction, Analog data, Digital data, Sampling, Sample, Sampling rate, Compression, Lossless compression, Lossy compression, Metadata, Data cleaning, Pattern, Correlation, Causation, Data bias, Incomplete data, Bar chart, Line graph, Pie chart, Scatter plot, Histogram, Truncated axis, Cherry-picking, Re-identification, Aggregation, Privacy.

About this guide. Written for Rycal and aligned to the College Board AP Computer Science Principles course framework, Unit 2. All questions and explanations are original Rycal writing. Rycal is independent and is not affiliated with or endorsed by the College Board.