UFV

UFV.ES

Information and Data

MODULE 3 / UNIT 9 - 1 / 0.8

MOISES M. MARTINEZ

FUNDAMENTALS OF COMPUTER ENGINEERING

2024/25

What is Data?

What is Information?

Are Data and Information the same thing?

Data vs Information

Data vs Information

Data consists of unorganized and unrefined raw facts.

Information is the organized and interpreted form of those raw facts (context).

Data vs Information

In the field of Computer Science, both concepts are fundamental, yet they are distinguished by significant differences:

Data	Information
Data refers unprocessed, raw facts lacking specific meaning.	Information refers processed data to imbue it with purpose and significance (context).
Data is independent of the information.	Information is dependent on data.
Raw data, in isolation, does not provide an adequate basis for decision-making.	Information, typically, furnishes sufficient context for informed decision-making.

Information in computers

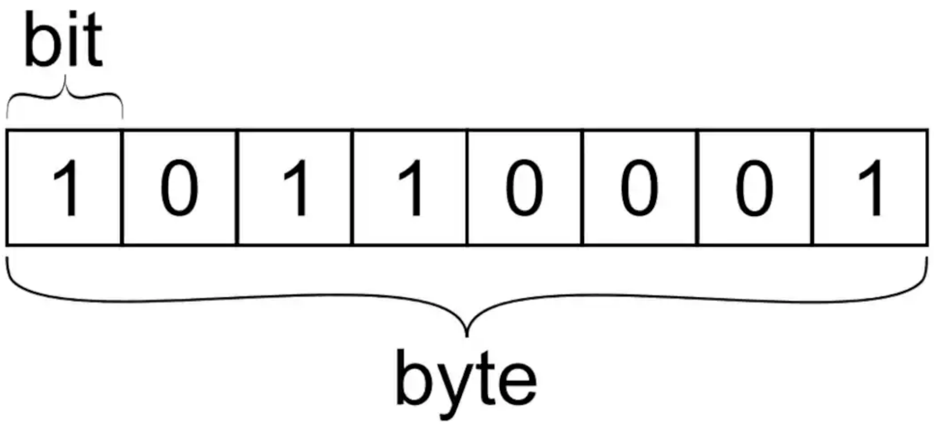
01

Information in computers

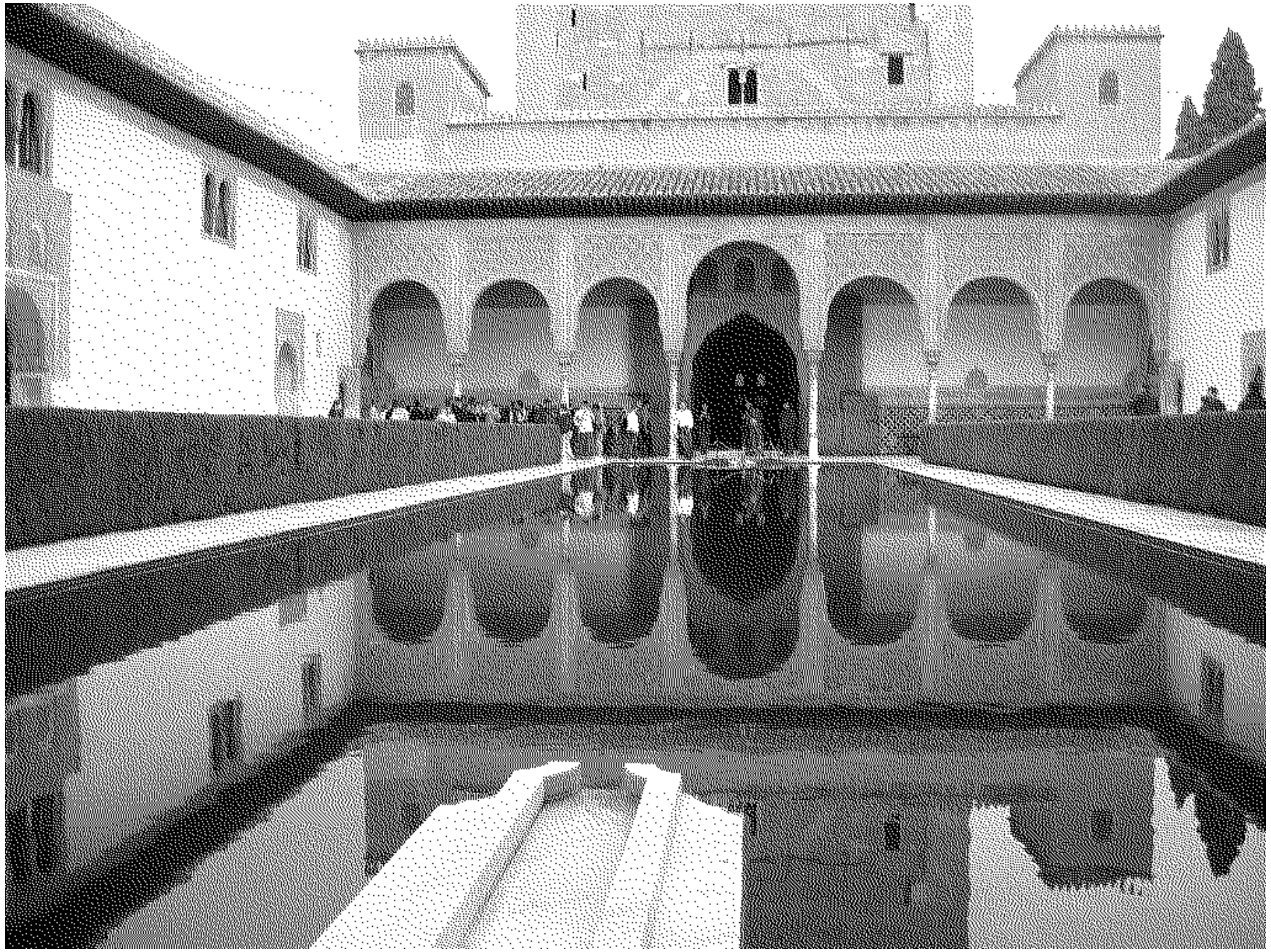
A bit, short for binary digit, is the **smallest elemental unit of data** in computing. A bit can represent only one of two distinct values: 0 or 1.

A byte is the smallest memory unit that can be accessed and utilized in most computer systems, often used to store data smaller than a byte in size. Depending on the context, a byte can serve as a container for various types of information, such as:

- Letters.
- A numbers.
- Program instructions.
- Pixels in an image or part of an audio recording.



Information in computers



Picture was created using 1 bit: 2 colors.

Information in computers

Information is represented using a grayscale model, where each pixel can display a specific shade of grey ranging from 0 to 255.

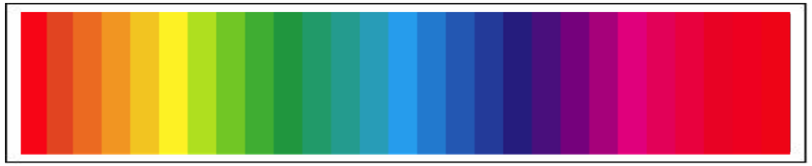
Black: 00000000

White: 11111111

Picture was created using gray-scale format: 256 unique colors in total.

Information in computers

Information is represented using multiple color layers (Red, Green, and Blue), along with an additional layer dedicated to the luminosity factor.



Black: 00000000

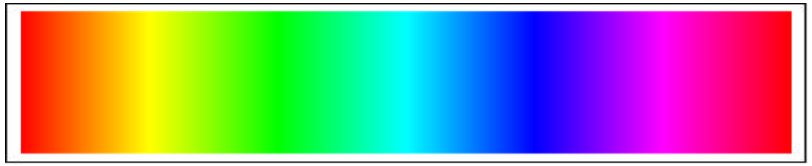
White: 11111111

16.7 million colors

Picture was created using 8-bit format: 256 unique colors per channel.

Information in computers

Information is represented using multiple color layers (Red, Green, and Blue), along with an additional layer dedicated to the luminosity factor.



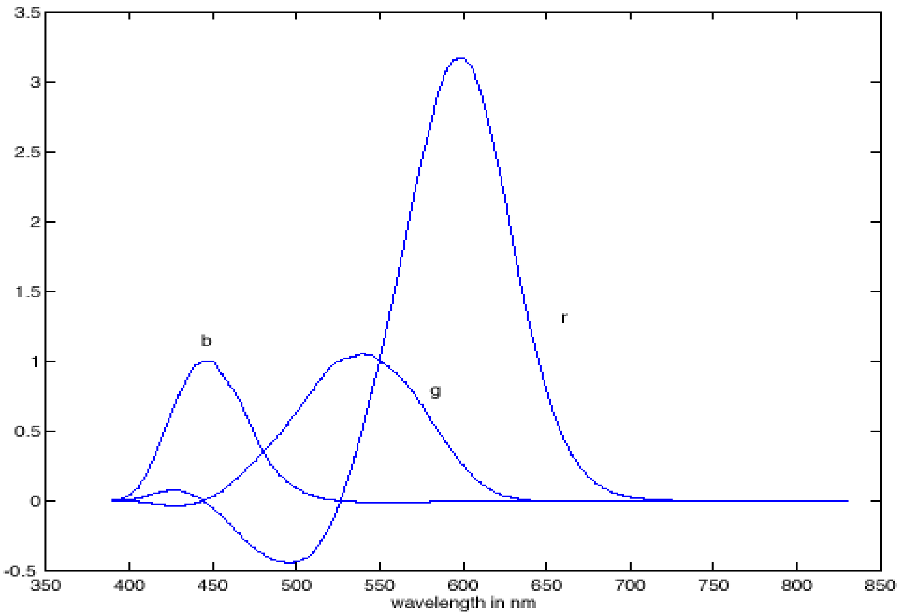
Black: 00000000

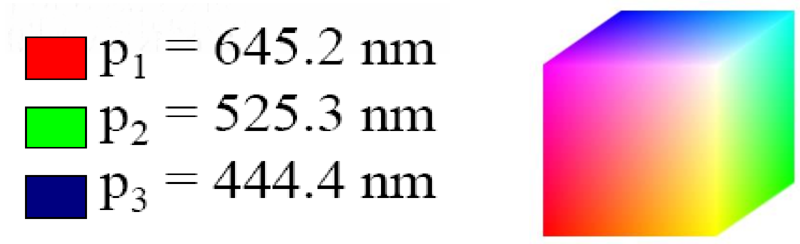
White: 11111111

281 trillion colors

Picture was created using 8-bit format: 256 unique colors per channel.

The RGB color model is an additive color model that combines red, green, and blue light in varying proportions to accurately reproduce a broad spectrum of colors. The name **RGB** nomenclature derive from the initials of the three primary colors used in this model: Red, Green, and Blue. By adjusting the intensity of each of these colors, a wide range of colors can be created, making the RGB model fundamental in digital imaging, displays, and various other technologies.





Hex representation

- #FF0000
- #00FF00
- #0000FF
- #FFFF00
- #CCEEFF

Each color in the RGB model is represented by a hexadecimal number consisting of 2 digits.

Numeral systems

02

Numeral systems

A numeral system, also known as a number system, is a formalized method of notation used to express numerical values. This system provides a mathematical framework for symbolically representing numbers from a predefined set, using consistent digits or symbols in an organized manner. Numeral systems are designed with specific goals:

- Representation: To represent a relevant and practical set of numbers within a particular context.
- Uniqueness: To assign each number within this set a unique or, at the very least, a standardized representation.
- Reflection of properties: To reflect and convey the algebraic and arithmetic properties and relationships inherent to the numbers themselves.

The base, or radix, of a numeral system denotes the quantity of unique digits or elements available in that system.

For example:

- **Decimal System**: Base = 10, Digits = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
- **Hexadecimal System**: Base = 16, Digits = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}

Numeral systems

The value of a number N in base b is given by:

$$N = d_n \times b^n + d_{n-1} \times b^{n-1} + \cdots + d_1 \times b^1 + d_0 \times b^0$$

where:

- N is the number being represented.
- b is the base or radix of the numeral system.
- d_{n-1}, …, d_1, d_0 are the digits of the number in base b, with each d_i being an integer such that $0 \leq d_i < b$.
- d_n is the most significant digit, and d_0 is the least significant digit.

Each digit d_i is multiplied by the base raised to the power of its position index i, starting from 0 for the least significant digit (rightmost position) to n for the most significant digit (leftmost position).

Numeral systems

The value of a number N in base b is given by:

$$N = d_n \times b^n + d_{n-1} \times b^{n-1} + \cdots + d_1 \times b^1 + d_0 \times b^0 + d_{-1} \times b^{-1} + d_{-2} \times b^{-2} + \cdots + d_{-m} \times b^{-m}$$

where b is the base of the number system (e.g., 2, 8, 10, or 16), d_i represents the digit at position i and d_i is a digit that ranges from 0 to b−1.

$$(352.45)_{10} = 3 \times 10^2 + 5 \times 10^1 + 2 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2}$$

$$= 3 \times 100 + 5 \times 10 + 2 \times 1 + 4 \times 0{,}1 + 5 \times 0{,}01$$

Numeral systems

Base conversion refers to the process of transforming a numerical value N from one base representation to another. This is typically done by expressing the number in its **polynomial notation**, which highlights the digits and their corresponding values within the original base s. Once in this form, the necessary mathematical operations are performed according to the rules and conventions of the base s.

$$N1 = (10101)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (21)_{10}$$

$$N2 = (14)_{16} = 1 \times 16^1 + 4 \times 16^0 = (20)_{10}$$

Numeral systems

A positional numeral system is a mathematical framework used to represent numbers through an ordered sequence of numeral symbols, commonly known as digits. In this system, the value of each symbol depends not only on the symbol itself but also on its position within the sequence, with each position corresponding to a specific power of the base.

- Binary system represents information using digits ranging from 0 to 1.

N =	1	0	1	1
B =	3	2	1	0

Numeral systems

A positional numeral system is a mathematical framework used to represent numbers through an ordered sequence of numeral symbols, commonly known as digits. In this system, the value of each symbol depends not only on the symbol itself but also on its position within the sequence, with each position corresponding to a specific power of the base.

- Decimal system represents information using digits ranging from 0 to 9.

N =	4	2	1	4
B =	3	2	1	0

Numeral systems

A positional numeral system is a mathematical framework used to represent numbers through an ordered sequence of numeral symbols, commonly known as digits. In this system, the value of each symbol depends not only on the symbol itself but also on its position within the sequence, with each position corresponding to a specific power of the base.

- Hexadecimal system represents information using digits ranging from 0 to 9 and letter from A to F.

N =	A	F	7	1
B =	3	2	1	0

From binary to decimal

03

From binary to decimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

What is the procedure for converting the n_2 number represented in base 2 to its equivalent representation in base 10?

$$n_2 = 11001010$$

From binary to decimal

Positional (numeral) systems

Step 1 - Count the quantity of digits contained within our binary numeral.

- Binary
- Decimal
- Hexadecimal
- Octal

$$\left.\begin{array}{c}1\\1\\0\\0\\1\\0\\1\\0\end{array}\right\}\ n = 8$$

From binary to decimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Perform a multiplication operation, proceeding from left to right, wherein each digit is multiplied by the power of two that corresponds to its position, commencing with 2^{n-1}

$$\left.\begin{matrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{matrix}\right\} \mathbf{n = 8}$$

From binary to decimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 3 - Aggregate the resultant values obtained from the aforementioned calculations.

7	6	5	4	3	2	1	0	
1×2^7								1
	1×2^6							1
		0×2^5						0
			0×2^4					0
				1×2^3				1
					0×2^2			0
						1×2^1		1
							0×2^0	0
128	64	0	0	8	0	2	0	

$n = 8$

From binary to decimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 3 - Aggregate the resultant values obtained from the aforementioned calculations.

7	6	5	4	3	2	1	0	
1×2^7								1
	1×2^6							1
		0×2^5						0
			0×2^4					0
				1×2^3				1
					0×2^2			0
						1×2^1		1
							0×2^0	0
128	64	0	0	8	0	2	0	

n = 8

= 202

From binary to octal

04

From binary to octal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

What is the procedure for converting the n_2 number represented in base 2 to its equivalent representation in base 8?

$$n_2 = 100111011101111$$

From binary to octal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 1 - Segment the bits into clusters of three, progressing from right to left.

$\boxed{100}\ \boxed{111}\ \boxed{011}\ \boxed{101}\ \boxed{111}$

Three-bit binary numbers range from 000 to 111, corresponding to the decimal values 0 through 7.

From binary to octal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every triplet into its corresponding single-digit octal representation.

$\boxed{100}\ \boxed{111}\ \boxed{011}\ \boxed{101}\ \boxed{111}$

From binary to octal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every triplet into its corresponding single-digit octal representation.

$\boxed{100}\ \boxed{111}\ \boxed{011}\ \boxed{101}\ \boxed{111}$

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to octal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every triplet into its corresponding single-digit octal representation.

$\boxed{100}\,\boxed{111}\,\boxed{011}\,\boxed{101}\,\boxed{111}$

7

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to octal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every triplet into its corresponding single-digit octal representation.

100	111	011	101	111
4	**7**	**3**	**5**	**7**

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to Hexadecimal

05

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

What is the procedure for converting the n_2 number represented in base 2 to its equivalent representation in base 16?

$$n_2 = 1001110111101111$$

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 1 - Segment the bits into clusters of four, progressing from right to left.

$\boxed{\ \ 100}\ \boxed{1110}\ \boxed{1110}\ \boxed{1111}$

Four-bit binary numbers range from 0000 to 1111, corresponding to the values 0 through F.

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every cluster into its corresponding single-digit octal representation.

$\boxed{100}\ \boxed{1110}\ \boxed{1110}\ \boxed{1111}$

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every cluster into its corresponding single-digit octal representation.

$\boxed{100}\ \boxed{1110}\ \boxed{1110}\ \boxed{1111}$

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every cluster into its corresponding single-digit octal representation.

$\boxed{100}\ \boxed{1110}\ \boxed{1110}\ \boxed{1111}$

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every cluster into its corresponding single-digit octal representation.

$\boxed{100}\ \boxed{1110}\ \boxed{1110}\ \boxed{1111}$

F

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every cluster into its corresponding single-digit octal representation.

100	1110	1110	1111
4	**E**	**E**	**F**

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 - Translate every cluster into its corresponding single-digit octal representation.

100	1110	1110	1111
4	**E**	**E**	**F**

Dec	Hex	Oct	Bin
0	0	000	0000
1	1	001	0001
2	2	002	0010
3	3	003	0011
4	4	004	0100
5	5	005	0101
6	6	006	0110
7	7	007	0111
8	8	010	1000
9	9	011	1001
10	A	012	1010
11	B	013	1011
12	C	014	1100
13	D	015	1101
14	E	016	1110
15	F	017	1111

It is important to note that numerical values in the range of 10 to 15 are represented by the capital letters A through F.

From decimal to binary

06

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

What is the procedure for converting the n_{10} number represented in base 10 to its equivalent representation in base 2?

$$n_{10} = 233$$

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 1 - Perform division by the desired target base and retain the remainder.

233 **/ 2**

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 1 - Perform division by the desired target base and retain the remainder.

233 / 2

$\boxed{1}$ 116

remainder

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 1 - Perform division by the desired target base and retain the remainder.

233 / **2**

$\boxed{1}$ $\boxed{116}$ updated target base

remainder

UFV

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 to n - we iteratively execute the division operation with the updated target base and retain the remainder until the new target base is reduced to 1.

233		/ 2
1	116	

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 to n - we iteratively execute the division operation with the updated target base and retain the remainder until the new target base is reduced to 1.

233								/ 2
1	116							/ 2
	0	58						/ 2
		0	29					/ 2
			1	14				/ 2
				0	7			/ 2
					1	3		/ 2
						1	1	/ 2
							1	

From binary to hexadecimal

Positional (numeral) systems

- Binary
- Decimal
- Hexadecimal
- Octal

Step 2 to n - we iteratively execute the division operation with the updated target base and retain the remainder until the new target base is reduced to 1.

233								/ 2
1	116							/ 2
	0	58						/ 2
		0	29					/ 2
			1	14				/ 2
				0	7			/ 2
					1	3		/ 2
						1	1	/ 2
							1	
1	0	0	1	0	1	1	1	

10010111 is 233 in decimal base?

10010111 is 233 in decimal base?

NO, 10010111 is 151.

From binary to hexadecimal

Positional (numeral) systems

Step n + 1 - Invert the order of the obtained remainders

- Binary
- Decimal
- Hexadecimal
- Octal

233								/ 2
1	116							/ 2
	0	58						/ 2
		0	29					/ 2
			1	14				/ 2
				0	7			/ 2
					1	3		/ 2
						1	1	/ 2
							1	
1	0	0	1	0	1	1	1	

From binary to hexadecimal

Positional (numeral) systems

Step n + 1 - Invert the order of the obtained remainders

- Binary
- Decimal
- Hexadecimal
- Octal

233								/ 2
1	116							/ 2
	0	58						/ 2
		0	29					/ 2
			1	14				/ 2
				0	7			/ 2
					1	3		/ 2
						1	1	/ 2
							1	
1	0	0	1	0	1	1	1	

Binary addition

07

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

	1	0	1	0	0
+	1	1	1	1	0

Binary addition is carried out in a **right-to-left** manner.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

	1	0	1	0	0
+	1	1	1	1	0
					0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

	1	0	1	0	0
+	1	1	1	1	0
				1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

Carry

		1			
	1	0	1	0	0
+	1	1	1	1	0
			0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

The carry is preserved to be used in the calculation of the next digit.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

Carry

		1			
	1	0	1	0	0
+	1	1	1	1	0
		1	0	1	0

An additional operation must be conducted, taking into account the carry from previous calculations.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

		Carry 1			
	1	0	1	0	0
+	1	1	1	1	0
		1 + 1	0	1	0

An additional operation must be conducted, taking into account the carry from previous calculations.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

	Carry				
	1				
	1	0	1	0	0
+	1	1	1	1	0
		0	0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

An additional operation must be conducted, taking into account the carry from previous calculations.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

Carry					
1					
	1	0	1	0	0
+	1	1	1	1	0
	0	0	0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

Carry

1					
	1	0	1	0	0
+	1	1	1	1	0
	0 + 1	0	0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

Carry					
1					
	1	0	1	0	0
+	1	1	1	1	0
	0 + 1	0	0	1	0

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

Carry

1					
	1	0	1	0	0
+	1	1	1	1	0
	0 + 1	0	0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as "10".

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

Carry

1					
	1	0	1	0	0
+	1	1	1	1	0
	1	0	0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

	1	0	1	0	0
+	1	1	1	1	0
1	1	0	0	1	0

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

The final carry is included if its value is 1. This result in an increase in the number of bits required to represent the sum.

Binary addition

Binary addition is similar to decimal addition but uses only the binary digits 0 and 1, but the process follows specific rules for combining these digits.

0	+	0	=	0
1	+	0	=	1
0	+	1	=	1
1	+	1	=	0 and carry 1.

	1	0	1	0	0	= 20
+	1	1	1	1	0	= 30
1	1	0	0	1	0	= 50

1 + 1 = 2. In the binary numeral system, the representation of the number 2 consists of two binary digits, which is denoted as “10”.

Numeral properties

08

Numeral properties

Some important questions ……

How many values can be represented by n bits in binary numeral system?

Numeral properties

Some important questions ……

How many values can be represented by n bits in binary numeral system? 2^n

What is the minimum number of bits required to represent

a total of m distinct values?

Numeral properties

Some important questions ……

How many values can be represented by n bits in binary numeral system? 2^n

What is the minimum number of bits required to represent a total of m distinct values?

$Log_2(n)$ by excess

$Log_2(91) = 6.50779 = 7$

Numeral properties

Some important questions ……

How many values can be represented by n bits in binary numeral system? 2^n

What is the minimum number of bits required to represent a total of m distinct values?

$Log_2(n)$ by excess
$Log_2(91) = 6.50779 = 7$

When employing n bits, assuming that the **minimum representable value** corresponds to **0**, what is the maximum achievable numerical value?

Numeral properties

Some important questions ……

How many values can be represented by n bits in binary numeral system? 2^n

What is the minimum number of bits required to represent a total of m distinct values?

$Log_2(n)$ by excess

$Log_2(91) = 6.50779 = 7$

When employing n bits, assuming that the **minimum representable value** corresponds to **0**, what is the maximum achievable numerical value? $2^n - 1$

Numeral properties

The use of calculators is prohibited during the exam.

$$2^0 = 1$$
$$2^1 = 2$$
$$2^2 = 4$$
$$2^3 = 8$$
$$2^4 = 16$$
$$2^5 = 32$$
$$2^6 = 64$$
$$2^7 = 128$$
$$2^8 = 256$$

Numeral properties

Decimal	Binary	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Integer numbers

09

Integer numbers

Integer values in computer systems are represented using a fixed number of bits. The range of integers that can be effectively represented depends on the bit width and the specific method of representation used.

- The representable range of natural numbers (unsigned integers) is $[0, 2^n-1]$, where n represents the bit width. This notation is referred to as n-bit unsigned.

> With a bit width of n = 4, natural numbers can be represented within the interval [0, 15]. This means that a 4-bit unsigned integer can hold values ranging from 0 to 15.

It is important to note that not all possible integer values can be represented; only those that fall within the defined range for the given bit width and representation method are accommodated.

Integer numbers

In binary systems, integers can be represented using different methods depending on specific requirements:

- Unsigned integers
- Signed integers
  - sign-magnitude
  - one's complement
  - two's complement
- Excess (or biased) integers

$$(1\quad 0\quad 1\quad 1\,.\,0\quad 1)_{10}$$

$$(1\quad 1\quad 0\quad 0\,.\,1\quad 1)_{10}$$

Integer numbers

Sign-magnitude notation

The sign-magnitude notation is a method of representing signed integers including a symbol, which represent its sign, along with its magnitude indicating whether the number is positive or negative.

In binary, the sign of a number can be represented using the Most Significant Bit (MSB): 0 (+) and 1 (-).

$(+16)_{10}$ = 00010000 in binary using 8-bit signed (Sign-magnitude notation)

$(-16\ )_{10}$ = 10010000 in binary using 8-bit signed (Sign-magnitude notation)

The range of numbers that can be represented using n bits: [$-(2^{n-1} - 1), (2^{n-1} - 1)$].

If we use 8-bit signed integers in sign-magnitude, the range of representable integer numbers is [−127,+127].

Integer numbers

Sign-magnitude notation

To compute the integer value of a signed binary number, follow these steps:

- Convert the magnitude to base 10 using the n−1 less significant bits (all bits except the most significant bit).
- Determine and apply the sign: (1) If the most significant bit (MSB) is 0, the number is positive. If the MSB is 1, the number is negative.

Example: $0\boxed{0111} = 7, \quad 1\boxed{1010} = -10$

Addition and subtraction operations in binary can be more complex because the sign and magnitude (highlighted in red) must be handled separately.

Zero has two possible representations: [−0, +0].

Integer numbers

One's complement notation

One's complement is a method for representing signed integers in binary, allowing both positive and negative numbers to be represented.

- The range of numbers that can be represented using n bits: $[-(2^{n-1} - 1), (2^{n-1} - 1)]$.
- The most significant bit (MSB) serves as the sign bit, where 0 indicates a positive number and 1 indicates a negative number.
- The One's complement notation has two representations for zero: +0 (all bits 0) and −0 (all bits 1).

One's complement notation

How One’s complement works for signed binary number:

- If the number is positive:
  1. Convert the decimal number into a binary number.
- If the number is negative:
  1. Remove the sign and use the positive number
  2. Convert the decimal number into a binary number.
  3. Invert all the bits (change 0s to 1s and 1s to 0s) to obtain the one's complement of the number.

$(+5)_{10}$ = 00000101 in binary using 8-bit signed (One's complement notation)

$(-5)_{10}$ = 11111010 in binary using 8-bit signed (One's complement notation)

One's complement notation

How do you convert 229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?**

One's complement notation

How do you convert 229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **8 bits**

One's complement notation

How do you convert 229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **8 bits**
- **How many bits I need to represent 229 in binary one’s complement?**

Integer numbers

One's complement notation

How do you convert 229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **8 bits**
- **How many bits I need to represent 229 in binary one’s complement?** **9 bits**

One's complement notation

How do you convert 229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **8 bits**
- **How many bits I need to represent 229 in binary one's complement?** **9 bits**

$$229 = 11100101 \rightarrow 11100101 \neq C1_2(11100101)$$

If we use an 8-bit binary representation, we can only represent integers within the range [−127, +127], so the number 229 falls outside of this range and cannot be represented.

Integer numbers

One's complement notation

How do you convert 229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **8 bits**
- **How many bits I need to represent 229 in binary one's complement?** **9 bits**

$$229 = 11100101 \rightarrow 011100101 = C1_2(11100101)$$

If we use an 8-bit binary representation, we can only represent integers within the range [−127, +127], so the number 229 falls outside of this range and cannot be represented.

However, if we expand to a 9-bit binary representation, we can represent integers within the range [-255, +255]. This allows us to represent 229, as it falls within this expanded range.

Integer numbers

One's complement notation

Decimal signed number	Positive binary	Negative binary
0	0000	1111
1	0001	1110
2	0010	1101
3	0011	1100
4	0100	1011
5	0101	1010
6	0110	1001
7	0111	1000

If we use a 4-bit binary representation allows for the representation of integer numbers in the range of [−7, 7] in one's complement notation. This range is determined by the fact that **the MSB is used to indicate the sign**, leaving the remaining 3 bits to represent the magnitude.

Integer numbers

Two's complement notation

Two's complement is a method for representing signed integers in binary, which allows both positive and negative numbers to be represented efficiently.

- The range of numbers that can be represented using n bits: $[-2^{n-1}, 2^{n-1} - 1]$.
- The most significant bit (MSB) serves as the sign bit, where 0 indicates a positive number and 1 indicates a negative number.
- The two's complement notation has only one representation for zero.

$$C_b(N) = b^n - N$$

If we use 4-bit signed → $C_{10}(129) = 10^4 - 129 = 9871$

$9871 + 129 = 10000 = 10^4$

Two's complement notation

How One’s complement works for signed binary number:

- If the number is positive:
  1. Convert the decimal number into a binary number.
- If the number is negative:
  1. Remove the sign and use the positive number
  2. Convert the decimal number into a binary number.
  3. Invert all the bits (change 0s to 1s and 1s to 0s) to obtain the one's complement of the number.
  4. Add 1 to the least significant bit (LSB) of the one's complement result.

$(+5)_{10}$ = 00000101 in binary using 8-bit signed (Two's complement notation)

$(-5)_{10}$ = 11111011 in binary using 8-bit signed (Two's complement notation)

Two’s complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number -229 in the binary base?**

Two’s complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number -229 in the binary base?**

We cannot

Two’s complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **We cannot**
- **How many bits I need to represent -229 in binary one’s complement?**

Integer numbers

Two's complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **We cannot**
- **How many bits I need to represent -229 in binary one's complement?** **9 bits**

1. Remove the sign and use the positive number → 229

Integer numbers

Two's complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **We cannot**
- **How many bits I need to represent -229 in binary one's complement?** **9 bits**

1. Remove the sign and use the positive number → 229
2. $C_2(-229)$ = **0 1 1 1 0 0 1 0 1**

Integer numbers

Two’s complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **We cannot**
- **How many bits I need to represent -229 in binary two’s complement?** **9 bits**

1. Remove the sign and use the positive number $\longrightarrow$ 229
2. $C_2(-229)$ = **0 1 1 1 0 0 1 0 1**
3. $C_2(-229)$ **= 1 0 0 0 1 1 0 1 0** $\longleftarrow$ We must flip all bits.

Two's complement notation

How do you convert -229 from decimal to binary?

- **How many bits I need to represent decimal number 229 in the binary base?** **We cannot**
- **How many bits I need to represent -229 in binary one's complement?** **9 bits**

1. Remove the sign and use the positive number → 229
2. C_2(-229) = **0 1 1 1 0 0 1 0 1**
3. C_2(-229) **= 1 0 0 0 1 1 0 1 0** ← We must flip all bits.
4. C_2(-229) + 1 = **1 0 0 0 1 1 0 1 1** ← Add 1 to the least significant bit (LSB)

C_2(-229) = **1 0 0 0 1 1 0 1 1** it is -229 in two's complement.

Integer numbers

Two's complement notation

How convert from decimal to two's complement (Tips):

n = 4 bits C_2 (1100) = 0100

n = 8 bits C_2 (11011100) = 00100100

n = 8 bits C_2 (11001010) = 00110110

1. We find the MSB. The first 1 starting on the right.
2. We must flip all bits after the MSB.

$$\mathbf{N + C_b(N) = b^n}$$

n = 4 digits, N = 7500, $C_{10}(N) = 2500$

7500 + 2500 = 10000 with n = 4 digits, the result is 0000

N and $C_b(N)$ are opposites $\rightarrow C_b(N) \sim -N$

Integer numbers

Two’s complement notation

How convert from decimal to two’s complement (Tips):

- Positive numbers are represented as a magnitude and sign (therefore starting with 0).
- Negative numbers are represented as the two's complement of the corresponding positive number (starting with 1).

Standard positional representation of a number N in base b using **two's complement** is written as follow:

$$N = (a_{n-1}a_{n-2} \dots a_1a_0\ a_{-1} \dots a_{-m})_b$$

Integer numbers

Two’s complement notation

How do you convert -68 to binary?

- **How many bits I need to represent decimal number -68 in the binary base?**

Two’s complement notation

How do you convert -68 to binary?

- **How many bits I need to represent decimal number -68 in the binary base?** **8-bit signed**

I need 7 bits to represent 68 but I need another bit more to represent -68 getting a range [-128 + 127].

Integer numbers

Two’s complement notation

How do you convert -68 to binary?

- **How many bits I need to represent decimal number -68 in the binary base?** **8-bit signed**

1. Remove the sign and use the positive number → 68

Integer numbers

Two's complement notation

How do you convert -68 to binary?

- **How many bits I need to represent decimal number -68 in the binary base?** **8-bit signed**

1. Remove the sign and use the positive number → 68
2. Convert the decimal number into a binary number.

68							**/**	**2**	
0	**34**						**/**	**2**	
	0	**17**					**/**	**2**	
		1	**8**				**/**	**2**	
			0	**4**			**/**	**2**	
				0	**2**		**/**	**2**	
					0	**1**	**/**	**2**	
							/	**2**	
0	**0**	**1**	**0**	**0**	**0**	**1**	**/**	**2**	← Add extra 0 to have 8 bits.

Two's complement notation

How do you convert -68 to binary?

- **How many bits I need to represent decimal number -68 in the binary base?** **8-bit signed**

1. Remove the sign and use the positive number → 68
2. Convert the decimal number into a binary number → **68 = 01000100**
3. Invert all the bits → **68 = 01000100** ← This is the one's complement

Two’s complement notation

How do you convert -68 to binary?

- **How many bits I need to represent decimal number -68 in the binary base?** **8-bit signed**

1. Remove the sign and use the positive number → 68
2. Convert the decimal number into a binary number → **68 = 01000100**
3. Invert all the bits → **68 = 10111011** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **10111011 + 1 = 10111100**

$C_2(-68)$ = **1 0 1 1 1 1 0 0** it is -68 in two’s complement.

Two's complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?**

Integer numbers

Two's complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

I need 13 bits to represent 5,491 but I need another bit more to represent -5,491 getting a range [-8,192 +8,191].

UFV

Two’s complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491

Two’s complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

5491														/	2
1	2745													/	2
	1	1372												/	2
		0	686											/	2
			0	343										/	2
				1	171									/	2
					1	85								/	2
						1	42							/	2
							0	21						/	2
								1	10					/	2
									0	5				/	2
										1	2			/	2
											0	1	2	/	2
												1			
1	1	0	0	1	1	1	0	1	0	1	0	1			

Integer numbers

Two's complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491
2. Convert the decimal number into a binary number → **5,491 = 1 0 1 0 1 0 1 1 1 0 0 1 1**

I should add an extra zero, but I can wait until the end.

Integer numbers

Two’s complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491
2. Convert the decimal number into a binary number → **5,491 = 1 0 1 0 1 0 1 1 1 0 0 1 1**
3. Invert all the bits → **5,491 = 0 1 0 1 0 1 0 0 0 1 1 0 0** ← This is the one's complement

Integer numbers

Two’s complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491
2. Convert the decimal number into a binary number → **5,491 = 1 0 1 0 1 0 1 1 1 0 0 1 1**
3. Invert all the bits → **5,491 = 0 1 0 1 0 1 0 0 0 1 1 0 0** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **0 1 0 1 0 1 0 0 0 1 1 0 0 + 1 = 0 1 0 1 0 1 0 0 0 1 1 0 1**

UFV

Two's complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491
2. Convert the decimal number into a binary number → **5,491 = 1 0 1 0 1 0 1 1 1 0 0 1 1**
3. Invert all the bits → **5,491 = 0 1 0 1 0 1 0 0 0 1 1 0 0** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **0 1 0 1 0 1 0 0 0 1 1 0 0 + 1 = 0 1 0 1 0 1 0 0 0 1 1 0 1**

C_2(-5491) = **0 1 0 1 0 1 0 0 0 1 1 0 0** it is -5491 in two's complement.

Two's complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491
2. Convert the decimal number into a binary number → **5,491 = 1 0 1 0 1 0 1 1 1 0 0 1 1**
3. Invert all the bits → **5,491 = 0 1 0 1 0 1 0 0 0 1 1 0 0** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **0 1 0 1 0 1 0 0 0 1 1 0 0 + 1 = 0 1 0 1 0 1 0 0 0 1 1 0 1**

C_2(-5491) = **0 1 0 1 0 1 0 0 0 1 1 0 0** it is -5,491 in two's complement (NO).

UFV

Two’s complement notation

How do you convert -5,491 to binary?

- **How many bits I need to represent decimal number -5,491 in the binary base?** **13-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 5,491
2. Convert the decimal number into a binary number → **5,491 = 1 0 1 0 1 0 1 1 1 0 0 1 1**
3. Invert all the bits → **5,491 = 0 1 0 1 0 1 0 0 0 1 1 0 0** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **0 1 0 1 0 1 0 0 0 1 1 0 0 + 1 = 0 1 0 1 0 1 0 0 0 1 1 0 1**

C_2(-5491) = $\boxed{\mathbf{1}}$ **0 1 0 1 0 1 0 0 0 1 1 0 1** it is -5,491 in two’s complement.

I add an extra zero, the number using with 13 bits.

Two's complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?**

Two's complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 9,451

Two's complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

I need 14 bits to represent 9451 but I need another bit more to represent -9451 getting a range [-16,384 +16,383].

Integer numbers

Two’s complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

9451															/	2
1	4725														/	2
	1	2362													/	2
		0	1181												/	2
			1	590											/	2
				0	295										/	2
					1	147									/	2
						1	73								/	2
							1	36							/	2
								0	18						/	2
									0	9					/	2
										1	4				/	2
											0	2			/	2
												0	1	2	/	2
													1			
1	1	0	1	0	1	1	1	0	0	1	0	0	1			

Two's complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 9,451
2. Convert the decimal number into a binary number → **9,451 = 1 0 0 1 0 0 1 1 1 0 1 0 1 1**

Integer numbers

Two’s complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 9,451
2. Convert the decimal number into a binary number → **9,451 = 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1**

Two’s complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 9,451
2. Convert the decimal number into a binary number → **9,451 = 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1**
3. Invert all the bits **9,451 = 1 0 1 1 0 1 1 0 0 0 1 0 1 0 0** ← This is the one's complement

Integer numbers

Two’s complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 9,451
2. Convert the decimal number into a binary number → **9,451 = 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1**
3. Invert all the bits **9,451 = 1 0 1 1 0 1 1 0 0 0 1 0 1 0 0** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **1 0 1 1 0 1 1 0 0 0 1 0 1 0 0 + 1 = 1 0 1 1 0 1 1 0 0 0 1 0 1 0 1**

Two's complement notation

How do you convert -9,451 to binary?

- **How many bits I need to represent decimal number -9,451 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 9,451
2. Convert the decimal number into a binary number → **9,451 = 0 1 0 0 1 0 0 1 1 1 0 1 0 1 1**
3. Invert all the bits **9,451 = 1 0 1 1 0 1 1 0 0 0 1 0 1 0 0** ← This is the one's complement
4. Add 1 to the least significant bit (LSB) → **1 0 1 1 0 1 1 0 0 0 1 0 1 0 0 + 1 = 1 0 1 1 0 1 1 0 0 0 1 0 1 0 1**

$C_2(-9451)$ = **1 0 1 1 0 1 1 0 0 0 1 0 1 0 1** it is -9,451 in two's complement.

Integer numbers

Two’s complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?**

Two’s complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 13,351

Two's complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

I need 14 bits to represent 13,351 but I need another bit more to represent -13,351 getting a range [-16,384 +16,383].

Integer numbers

Two’s complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

13351															/	2
1	6675														/	2
	1	3337													/	2
		1	1668												/	2
			0	834											/	2
				0	417										/	2
					1	208									/	2
						0	104								/	2
							0	52							/	2
								0	26						/	2
									0	13					/	2
										1	6				/	2
											0	3			/	2
												1	1	2	/	2
													1			
1	1	1	0	0	1	0	0	0	0	1	0	1	1			

Two’s complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 13,351
2. Convert the decimal number into a binary number → **13,351 = 1 1 0 1 0 0 0 0 1 0 0 1 1 1**

Integer numbers

Two's complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 13,351
2. Convert the decimal number into a binary number → **13,351 = 0 1 1 0 1 0 0 0 0 1 0 0 1 1 1**

Remember, this is the most significant bit.

Integer numbers

Two's complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 13,351
2. Convert the decimal number into a binary number → **13,351 = 0 1 1 0 1 0 0 0 0 1 0 0 1 1 1**
3. Invert all the bits until the Most Significant Bit **13,351 = 1 0 0 1 0 1 1 1 1 0 1 1 0 0 0** ← This is the one's complement

Integer numbers

Two’s complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 13,351
2. Convert the decimal number into a binary number → **13,351 = 0 1 1 0 1 0 0 0 0 1 0 0 1 1 1**
3. Invert all the bits until the Most Significant Bit **13,351 = 1 0 0 1 0 1 1 1 1 0 1 1 0 0 0**
4. Add 1 to the least significant bit (LSB) → **1 0 0 1 0 1 1 1 1 0 1 1 0 0 0 + 1 = 1 0 0 1 0 1 1 1 1 0 1 1 0 0 1**

Integer numbers

Two's complement notation

How do you convert -13,351 to binary?

- **How many bits I need to represent decimal number -13,351 in the binary base?** **14-bit signed (+1 bit)**

1. Remove the sign and use the positive number → 13,351
2. Convert the decimal number into a binary number → **13,351 = 0 1 1 0 1 0 0 0 0 1 0 0 1 1 1**
3. Invert all the bits until the Most Significant Bit **13,351 = 1 0 0 1 0 1 1 1 1 0 1 1 0 0 0**
4. Add 1 to the least significant bit (LSB) → **1 0 0 1 0 1 1 1 1 0 1 1 0 0 0 + 1 = 1 0 0 1 0 1 1 1 1 0 1 1 0 0 1**

C_2(-13,351) = **1 0 0 1 0 1 1 1 1 0 1 1 0 0 1** it is -13,351 in two's complement.