Name: Answer Key	Date:
Topic: 1.0 Computer Number Systems	Period:

Word Bank

2	Expanded	place
10	Exponential	position
base	Factor	subscript
Binary	Hexadecimal	symbols
Decimal	how many	

Main Ideas	Notes
Number System	A positional number system defines: • which symbols are use • how many symbols are used • what each position in the number means Common Number Systems • Decimal Number System - Uses 10 symbols per digit • Binary Number System - Uses 2 symbols per digit • Hexadecimal Number System - Uses 16 symbols per digit The number of symbols allowed per digit is the base or radix of that number system. The base is written with a subscript such as 10011100_2. Decimal is a base 10 number system. Binary is a base 2 number system.

CFU 1.0.a

Place the following numbers in the number system that they could represent:

1. 01110010
2. FA084C
3. PQPQPP
4. 769213
5. YYNNYN

Binary	Decimal	Hexadecimal
1 3 5	4	2

What is a possible way to represent 16 symbols to make a base 16 number system?

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

How could you encode a base 36 number system?

0 ... 9, A ... Z

Reflection

Main Ideas	Notes
Expanded Form of Decimal Number	Expanded Form means that each digit is written as a separate number according to its place value.

The decimal number 683.57

Thousands	Hundreds	Tens	Ones		Tenths	Hundredths
	6	8	3	.	5	7

In Expanded Form is:

$$600 + 80 + 3 + 0.5 + 0.07$$

In Expanded Factor Form is:

$$6 \times 100 + 8 \times 10 + 3 \times 1 + 5 \times \frac{1}{10} + 7 \times \frac{1}{100}$$

In Expanded Exponential Form is:

$$6 \times 10^{2} + 8 \times 10^{1} + 3 \times 10^{0} + 5 \times 10^{-1} + 7 \times 10^{-2}$$

CFU 1.0.b

Identify the place value (hundreds, tens, ones, tenths, hundredths, etc.) of the digit in each of the following decimal numbers and write it in expanded form:

- 15.32, the 2 is in the hundredth place, in Expanded Factor Form expands to $2 \times \frac{1}{100}$
- 850.40, the 8 is in the hundreds place, in Expanded Exponential Form expands to 8×10^2

Main Ideas	Notes
Expanded Form in Binary	The expanded form of binary replaces the 10 in the exponent with a 2.

The binary number 1100.11_2

Eights	Fours	Twos	Ones		Halves	Quarters
1	1	0	0	.	1	1

In Expanded Form is:

$$1000_2 + 100_2 + 0.1_2 + 0.01_2$$

In Expanded Factor Form is:

$$1 \times 8 + 1 \times 4 + 1 \times \frac{1}{2} + 1 \times \frac{1}{4} = 12\frac{3}{4}$$

In Expanded Exponential Form is:

$$1 \times 2^{3} + 1 \times 2^{2} + 1 \times 2^{-1} + 1 \times 2^{-2}$$

CFU 1.0.c

Determine the values of the following binary numbers:

1. $1100.01_2 = 12.25_{10}$
2. $1001_2 = 9_{10}$
3. $1010.1_2 = 10.5_{10}$
4. $1111.11_2 = 15.75_{10}$

Eights	Fours	Twos	Ones		Halves	Quarters
1	1	0	0	.	0	1

Name:	Date:
Topic: 1.1 Conversion From Decimal To Binary	Period:

Word Bank

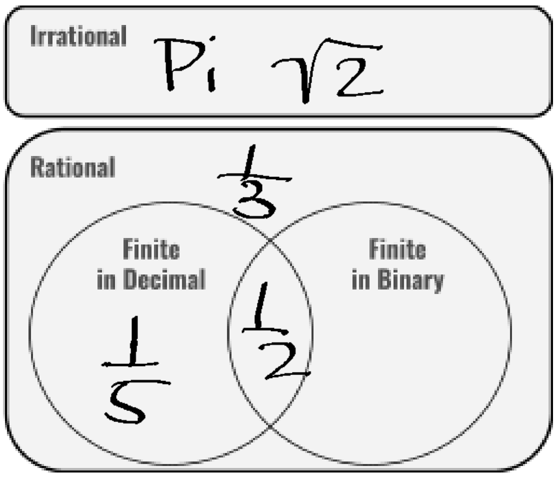
2
decomposing
denominator
finite
fractions
numerator
rational
zero

Main Ideas	Notes
Mismatched number systems	A rational number is any number that can be written as a simple fraction where both the top number (numerator) and the bottom number (denominator) are integers, and the bottom number is not zero. Not all rational numbers can be represented as a finite number (a number that can be written without repeating decimals). In **decimal**, only fraction with a denominator divisible by 2 or 5 are finite. In **binary**, only numbers with a denominator divisible by a power of 2 are finite.

CFU
1.1.a

Place the following numbers in the best category describing that number.

1. Pi
2. $\sqrt{2}$
3. 1/2
4. 1/3
5. 1/5



What might be a problem with not being able to represent amounts like $0.10 finitely in binary?

you will lose fractions of pennies due to round-off or truncation

Reflection

Main Ideas	Notes

Subtraction Method	Convert from decimal to binary by decomposing (breaking up) a number into all the binary place values that add up to your number.
	
	1 / 16 0 / 8 1 / 4 1 / 2 0 / 1
	
	1. Create a placeholder for all the binary place values that are less than your number.
	2. Find the largest number that is still less than your number.
	   a. place a one in that place holder
	   b. subtract that place value from your number
	3. Repeat step 2 until your number is zero and all your place holder values add up to your original number.
	Example: The number 22 can subtract 16, leaving 6, 6 can subtract 4 leaving 2. 22 = 16 + 4 + 2 = 10110_2

CFU 1.1.b	*Convert the following decimal numbers to their binary equivalent using the subtraction method:* 1) 17 $= 16 + 1 = 10001_2$ 2) 23 $= 16 + 4 + 2 + 1 = 10111_2$ 3) 57 $= 32 + 16 + 8 + 1 = 111001_2$ 4) 105 $= 64 + 32 + 8 + 1 = 1101001_2$

Remainder Method	Convert from decimal to binary by repeatedly dividing by the base number and using the remainder as the next digit (right-to-left) of the new number. **Example: The number 13 ÷ 2 = 6 remainder 1, 6 ÷ 2 = 3 remainder 0, 3 ÷ 2 = 1 remainder 1. Rebuild as 1101_2 = 8 + 4 + 1 = 13**

CFU 1.1.c	*Convert the following decimal numbers to their binary equivalent using the remainder method:* 5) 61 $= 111101_2 = 32 + 16 + 8 + 4 + 1$ 6) 133 $= 10000101_2 = 128 + 4 + 1$ 7) 253 $= 11111101_2$

Name:	Date:
Topic: 1.2 Adding in Binary	Period:

Word Bank

0	column
1	left
10	right
11	row

Main Ideas	Notes
Carrying (or Regrouping)	You can add directly in binary, if you follow the same procedure as in decimal. But it is helpful to note that: $0_2 + 0_2 = \underline{0}_2$ $0_2 + 1_2 = \underline{1}_2$ $1_2 + 1_2 = \underline{10}_2$ $1_2 + 1_2 + 1_2 = \underline{11}_2$ 1. Add one column at a time starting from the right column. 2. Carry any 2's digit. 3. Do this until you have reached the largest place value. Try it: $\begin{array}{r} 011010_2 \\ +\ 011011_2 \\ \hline 110101 \end{array}$ $\begin{array}{r} (26) \\ (27) \\ \hline (53) \end{array}$

CFU 1.2.a

Add the following number is binary (without converting to decimal for adding).

$$\begin{array}{r} 011010_2 \\ +010011_2 \\ \hline 101101 \end{array} \qquad \begin{array}{r} 111110_2 \\ +011011_2 \\ \hline 1011001 \end{array} \qquad \begin{array}{r} 111010_2 \\ +111001_2 \\ \hline 1110011 \end{array}$$

Name:	**Date:**
Topic: 1.3 Hexadecimal	**Period:**

Word Bank

16	C	F
A	D	nibble
B	digit	one
binary	divide	right
byte	E	zero

Main Ideas	**Notes**
Why hexadecimal?	All computer data is stored in groups of 8 bits called a byte. Every nibble (4 bits) of binary can directly translate to one digit of hexadecimal. It is easier to write the binary number 01101010_2 as $6A_{16}$ or 0x6A. The hexadecimal is a base-16 number and is a short-hand for binary data.
Hexadecimal to Decimal	Each place value is 16 times the place value to the right: **B 3 F** **256 16 1** **Hexadecimal Symbols:** 10 = A 11 = B 12 = C 13 = D 14 = E 15 = F In Expanded Factor Form: $B_{16} \times 256 + 3_{16} \times 16 + F_{16} \times 1$ In Expanded Factor Form (in decimal values): $11 \times 256 + 3 \times 16 + 15 \times 1$ In Expanded Form: $2816 + 48 + 15 = 2879$

CFU 1.3.a

Convert the following hexadecimal numbers to decimal:

$9F_{16}$	ACE_{16}	$F07_{16}$
$9 \times 16 + 15 \times 1$	$10 \times 256 + 12 \times 16 + 14$	$15 \times 256 + 7$
159	2,766	3,847

Main Ideas	Notes
Decimal to Hexadecimal 0.1875 = 3/16 0.4375 = 7/16	Use the remainder method to convert decimal to hexadecimal by using the remainder to continuously divide by 16. Example: 41,959 0 r 10 16 ⟌10 r 3 16 ⟌163 r 14 16 ⟌2622 r 7 16 ⟌41959 Translate the 4 values 10, 3, 14 and 7 to their hexadecimal symbols: **$A3E7_{16}$**
CFU 1.3.b	*Convert the following decimal numbers to their hexadecimal equivalent using the remainder method:* **9456** 2 r 4 16 ⟌36 r 15 = F_{16} 16 ⟌591 r 0 16 ⟌9456 $24F0_{16}$ **80,933,479** 4D2F267
Binary-to-Hexadecimal Hexadecimal-to-Binary	Because a nibble can be 0000 - 1111 which is 0 through 15, a binary number can be grouped in 4's from the 1's place (right-to-left) to translate to hexadecimal. **<u>01</u> <u>1010</u> <u>1101</u> <u>0011</u>** 1 10 13 3 1 A D 3
CFU 1.3.c	*Convert the following **binary** numbers to hexadecimal:* **0110101111000111** 6 B C 7 **1001011011000101** 9 6 C 5 *Convert the following **hexadecimal** numbers to binary:* **8FB3** 1000 1111 1011 0011 **E28C** 1110 0010 1000 1100

Name:	**Date:**
Topic: 1.4 Encoding Information with Binary	**Period:**

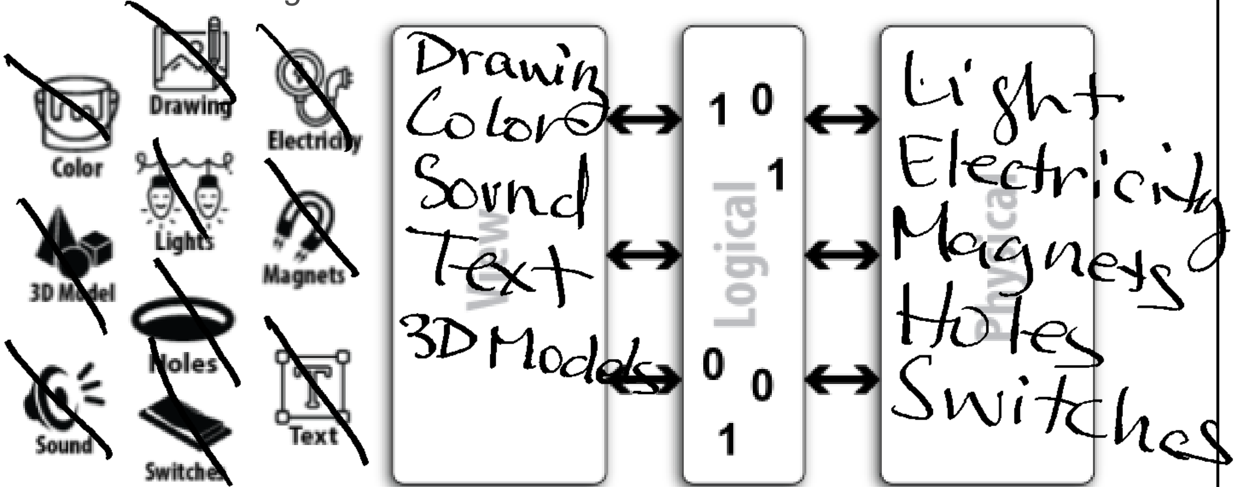
Word Bank

30	~~Characters~~	~~protocol~~
1024	~~Colors~~	~~Sounds~~
1,048,576	Drawings	~~View~~
~~binary~~	~~Logical~~	

Main Ideas	**Notes**
Logical Data Storage	**ALL** digital content is stored as a binary number. Each of the following View Layer items can be translated to its Logical Layer mapping. • Characters are mapped to a number using a standard called ASCII. ASCII is a protocol for encoding text. For example 'A' is stored as the binary number 01000001_2 which is the decimal value 65. • Color are mapped to a number by breaking up their composition into its red, green and blue components. • Sounds are mapped to numbers by storing their volume at small intervals so the waves can be reproduced. • Drawings, 3D Models and illustrations are mapped to numbers by storing the geometry vectors as numbers that describe them.

CFU 1.4.a

*Place items in the **Physical Layer** that can represent zeros and ones physically. Place items in the **View Layer** that can be encoded to be stored on a computer and retrieved to be viewed again.*



Main Ideas	Notes

Reflection

What benefit does having everything go through a logical layer of zeros and ones provide? All data can convert to binary. All binary can be stored in many physical devices

Storage Size

We typically think of a kilo representing 1,000 of something, but computer storage is measured in powers of 2.

Kilobyte = 1,024 (2^{10})

Megabyte = 1,048,576 (2^{20})

Gigabyte = 2 to the power of 2^{30} (approx. 1.073 billion)

CFU 1.4.b

Determine the amount of storage you need for each of the following scenarios.

1) I have to submit an essay for an application and they told me that it should be less than 2000 words. (Assume a word is an average of 6 characters and one character for the space.)

   ~12kbytes

2) I want a high-fidelity recording of a performance that will last for 5 minutes. Each sound point requires 4 bytes of data and the points are at 10 millisecond intervals. $5\,min \times \frac{60\,sec}{min} \times \frac{1000\,ms}{s} \times \frac{4\,bytes}{10\,ms}$

   ~120kbytes

Numeric Range of a binary number

# of bits	values	total unique values 2^n
1	`0, 1`	2
2	`00, 01, 10, 11`	$2^2 = 4$
4	`000, 0001, …, 1111`	$2^4 = 2 \times 2 \times 2 \times 2 = 16$
8	`00000000, … 11111111`	$2^8 = 256$

CFU 1.4.c

How many bits would I need to be able to store each of the following distinct values:

65,536 16 bits

16,777,216 24 bits

4,294,976,296 32 bits