Ingeniería de Software

Sistemas Numéricos – Algebra Booleana

Britney Tatiana Torres Ochoa

1021312652

Reyes Chaparro Joseph Imanol

1013261629

Prada Ariza Joseph Fabian

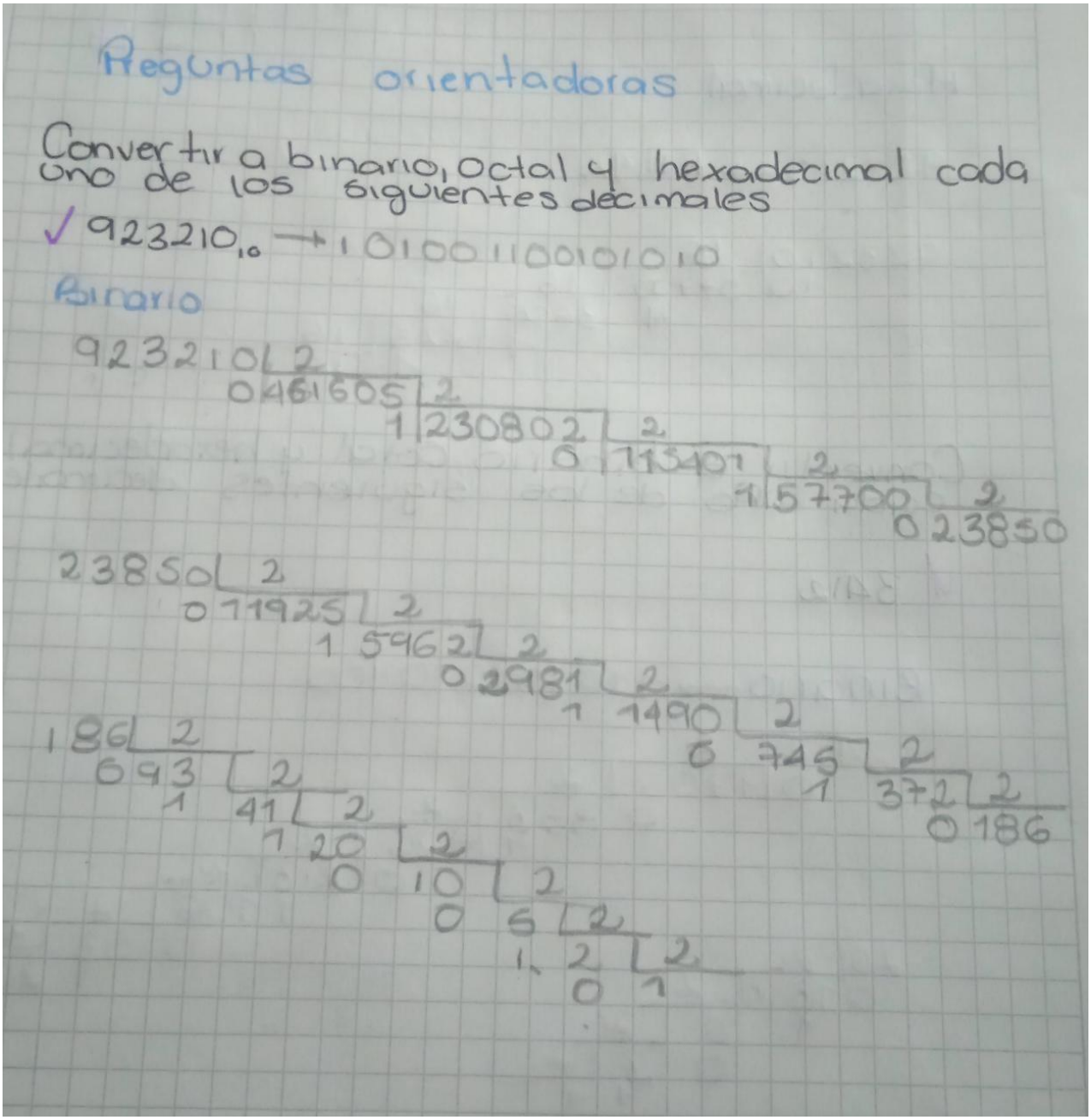
1018414514

Agosto de 2023

Preguntas Orientadoras

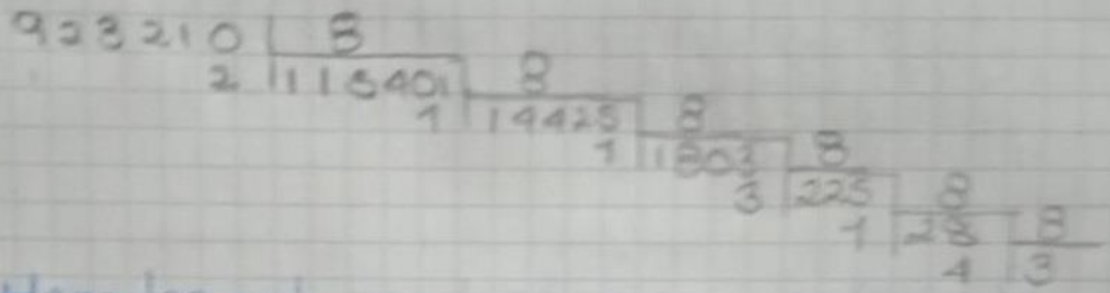
Convertir a binario, octal y hexadecimal cada uno de los siguientes decimales.

- A. 923210_{10}
- B. 3412_{10}
- C. 917_{10}



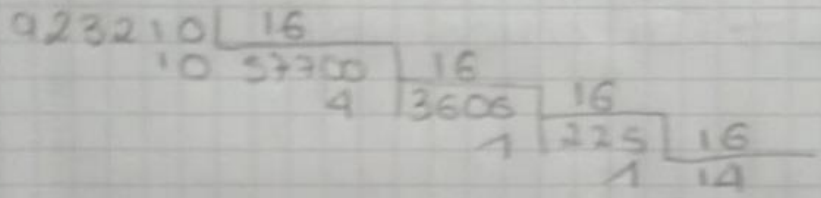
Octal →

923210 → 3413112



Hexadecimal

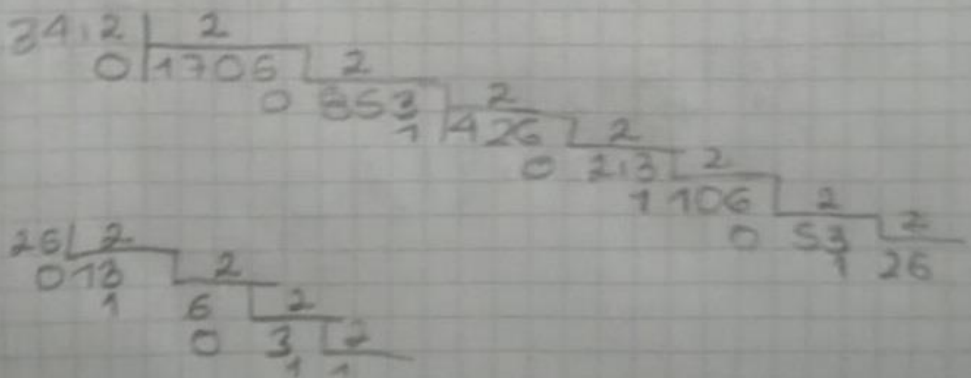
923210 → E114A



Convertir a binario, octal y hexadecimal cada uno de los siguientes decimales

✓ 3412

Binario → 110101010100



Octal

3A12 → 65514

3d12 | 8
4 426 | 8
2 53 | 8
5 6

Hexadecimal

3A12 → D.64

3A12 | 16
9 213 | 16
5 13

Convertir a binario, octal y hexadecimal cada uno de los siguientes decimales

✓ 917

Binario

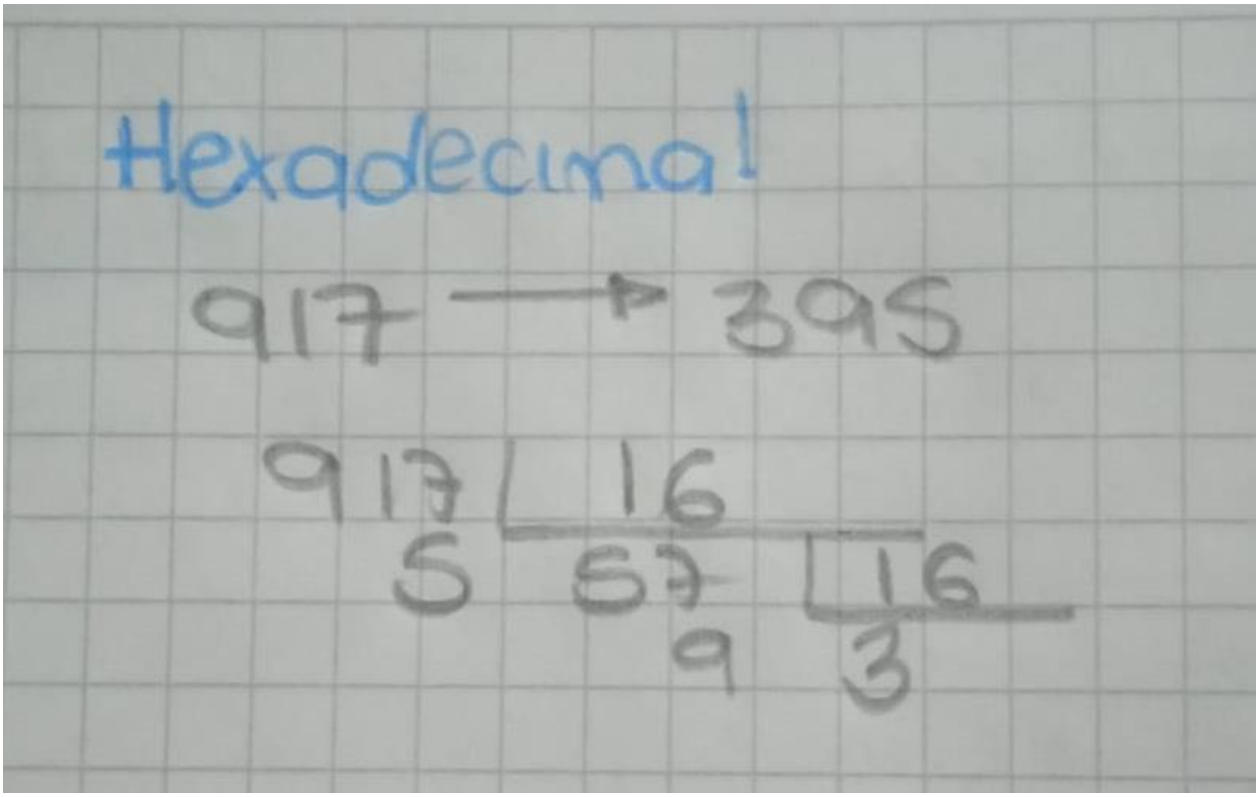
917 → 1110010101

917 | 2
1 458 | 2
0 229 | 2
1 114 | 2
0 57 | 2
1 28 | 2
0 14 | 2
0 7 | 2
1 3 | 2
1 1

Octal

917 → 1625

917 | 8
5 114 | 8
2 14 | 8
6 1


Hexadecima!
917 → 395
917 | 16
5 57 | 16
9 3

1. Realiza un video no mayor a 5 minutos que explique el proceso de conversión de los sistemas de numeración binario, decimal, octal hexadecimal.

https://youtu.be/iKONrZhMheU

2. Busca y toma una imagen de la tabla de código ASCII.

TABLA DE CARACTERES DEL CÓDIGO ASCII

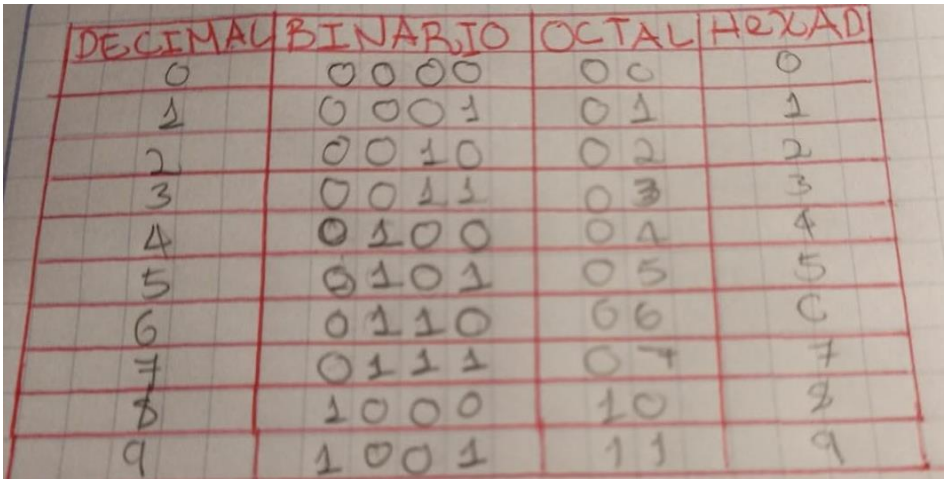
1 ☺	25 ↓	49 1	73 I	97 a	121 y	145 æ	169 ⌐	193 ┴	217 ┘	241 ±
2 ☻	26	50 2	74 J	98 b	122 z	146 Æ	170 ¬	194 ┬	218 ┌	242 ≥
3 ♥	27	51 3	75 K	99 c	123 {	147 ô	171 ½	195 ├	219 ▮	243 ≤
4 ♦	28 ∟	52 4	76 L	100 d	124 |	148 ö	172 ¼	196 ─	220 ▄	244 ⌠
5 ♣	29 ↔	53 5	77 M	101 e	125 }	149 ò	173 ¡	197 ┼	221 ▌	245 ⌡
6 ♠	30 ▲	54 6	78 N	102 f	126 ~	150 û	174 «	198 ╞	222 ▐	246 ÷
7	31 ▼	55 7	79 O	103 g	127 ⌂	151 ù	175 »	199 ╟	223 ▀	247 ≈
8	32	56 8	80 P	104 h	128 Ç	152 ÿ	176 ░	200 ╚	224 α	248 °
9	33 !	57 9	81 Q	105 i	129 ü	153 Ö	177 ▒	201 ╔	225 ß	249 ∙
10	34 "	58 :	82 R	106 j	130 é	154 Ü	178 ▓	202 ╩	226 Γ	250 ·
11	35 #	59 ;	83 S	107 k	131 â	155 ¢	179 │	203 ╦	227 π	251 √
12	36 $	60 <	84 T	108 l	132 ä	156 £	180 ┤	204 ╠	228 Σ	252 ⁿ
13	37 %	61 =	85 U	109 m	133 à	157 ¥	181 ╡	205 ═	229 σ	253 ²
14	38 &	62 >	86 V	110 n	134 å	158 ₧	182 ╢	206 ╬	230 µ	254 ■
15	39 '	63 ?	87 W	111 o	135 ç	159 ƒ	183 ╖	207 ╧	231 τ	255
16 ►	40 (	64 @	88 X	112 p	136 ê	160 á	184 ╕	208 ╨	232 Φ	
17	41)	65 A	89 Y	113 q	137 ë	161 í	185 ╣	209 ╤	233 Θ	
18 ↕	42 *	66 B	90 Z	114 r	138 è	162 ó	186 ║	210 ╥	234 Ω	
19 ‼	43 +	67 C	91 [	115 s	139 ï	163 ú	187 ╗	211 ╙	235 δ	
20 ¶	44 ,	68 D	92 \	116 t	140 î	164 ñ	188 ╝	212 ╘	236 ∞	
21 §	45 -	69 E	93]	117 u	141 ì	165 Ñ	189 ╜	213 ╒	237 φ	
22 ▬	46 .	70 F	94 ^	118 v	142 Ä	166 ª	190 ╛	214 ╓	238 ε	
23 ↨	47 /	71 G	95 _	119 w	143 Å	167 º	191 ┐	215 ╫	239 ∩	
24 ↑	48 0	72 H	96 `	120 x	144 É	168 ¿	192 └	216 ╪	240 ≡	

www.rey-dec.com

PRESIONA LA TECLA Alt MÁS EL NÚMERO

CORTESÍA DE: REYDEC desde 1976

3. Consulta y realiza la tabla de hexadecimal con cuatro entradas.



DECIMAL	BINARIO	OCTAL	HEXADI
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	C
7	0111	07	7
8	1000	10	8
9	1001	11	9

4. Consulta y explica con un ejemplo la Aritmética de punto fijo.

La aritmética de punto fijo es una técnica utilizada en ciertas aplicaciones de procesamiento digital de señales y sistemas embebidos donde los números se representan de manera fija con una cantidad predefinida de bits para la parte entera y la parte fraccionaria. A diferencia de la aritmética de punto flotante, en la que se utilizan exponentes y mantisas variables, la aritmética de punto fijo se basa en representar números con una escala fija.

Ejemplo:

Supongamos que queremos representar el número decimal 6.75 en una aritmética de punto fijo con 8 bits (4 bits para la parte entera y 4 bits para la parte fraccionaria).

Paso 1: Convertir la parte entera y fraccionaria a binario:

Parte entera: 6 en binario es 0110.

Parte fraccionaria: 0.75 en binario es 0.11.

Paso 2: Completar con ceros:

Parte entera: 0110 (ya tiene 4 bits).

Parte fraccionaria: 0.11, completamos con ceros para tener 4 bits en total: 1100.

Paso 3: Concatenar las partes:

La representación binaria del número 6.75 en aritmética de punto fijo de 8 bits sería: 0110.1100.

En este ejemplo, el número decimal 6.75 se ha representado en aritmética de punto fijo utilizando 8 bits, con 4 bits para la parte entera y 4 bits para la parte fraccionaria. Es importante tener en cuenta que la precisión de los cálculos y la representación dependerá de la cantidad de bits asignados a cada parte.

La aritmética de punto fijo es útil en aplicaciones donde el rango de valores y la precisión se conocen de antemano, ya que permite un control más preciso sobre el uso de recursos de hardware y simplifica los cálculos en comparación con la aritmética de punto flotante. Sin embargo, también tiene limitaciones en términos de rango y precisión en comparación con la aritmética de punto flotante.

5. Consulta y explica con un ejemplo la Aritmética de punto flotante.

La aritmética de punto flotante es una técnica utilizada para representar y realizar operaciones con números reales en sistemas computacionales. A diferencia de la aritmética de punto fijo, en la que se fija el número de bits para la parte entera y fraccionaria, en la aritmética de punto flotante se utiliza una representación que incluye una mantisa, un exponente y una base.

La representación general de un numero en punto flotante es: (-1)^S × M × $B\epsilon$, donde:

- S es el bit de signo (0 para positivo, 1 para negativo)
- M es la mantisa, que es una fracción normalizada entre 1 y la base B.
- E es el exponente, que ajusta el valor de la mantisa para mover el punto decimal.

En un sistema binario, como es común en las computadoras, la base $B\ es\ 2$

Ejemplo:

Supongamos que estamos usando una representación de punto flotante de 32 bits con 1 bit para el signo, 8 bits para el exponente y 23 bits para la mantisa (siguiendo el estándar IEEE 754 para números de precisión simple en punto flotante).

Deseamos representar el número decimal 12.75 en esta aritmética de punto flotante.

Paso 1: Convertir 12.75 a binario:

La parte entera de 12 en binario es 1100. La parte fraccionaria de 0.75 se puede convertir multiplicando sucesivamente por 2 y tomando las partes enteras: 0.75 * 2 = 1.5 (entero 1), 0.5 * 2 = 1.0 (entero 1). Por lo tanto, la representación binaria de 0.75 es 0.11.

Paso 2: Normalizar la mantisa: La mantisa normalizada será 1.100×231.100×2 , ya que hemos movido el punto decimal 3 lugares a la izquierda para que la parte entera sea 1.

Paso 3: Representar el exponente:

El exponente es 33, que se representa en binario como 0000001100000011.

Paso 4: Determinar el bit de signo: Como el número es positivo, el bit de signo es 00.

En conjunto, la representación en punto flotante del número 12.7512.75 sería:

0 |00000011|10000000000000000000000,0|00000011|10000000000000000000000,

donde el primer bit es el bit de signo, los siguientes 8 bits son el exponente, y los últimos 23 bits son la mantisa.

La aritmética de punto flotante es útil para manejar una amplia gama de valores y proporcionar una mayor precisión en comparación con la aritmética de punto fijo. Sin embargo, también introduce ciertos desafíos debido a la naturaleza de los cálculos con exponentes y mantisas variables.

6. Realiza el proceso de las siguientes conversiones:

Convertir a binario, octal y hexadecimal cada uno de los siguientes decimales.

✓ a. 325_{10} b. 954_{10} c. 1562_{10} d. 2463_{10}

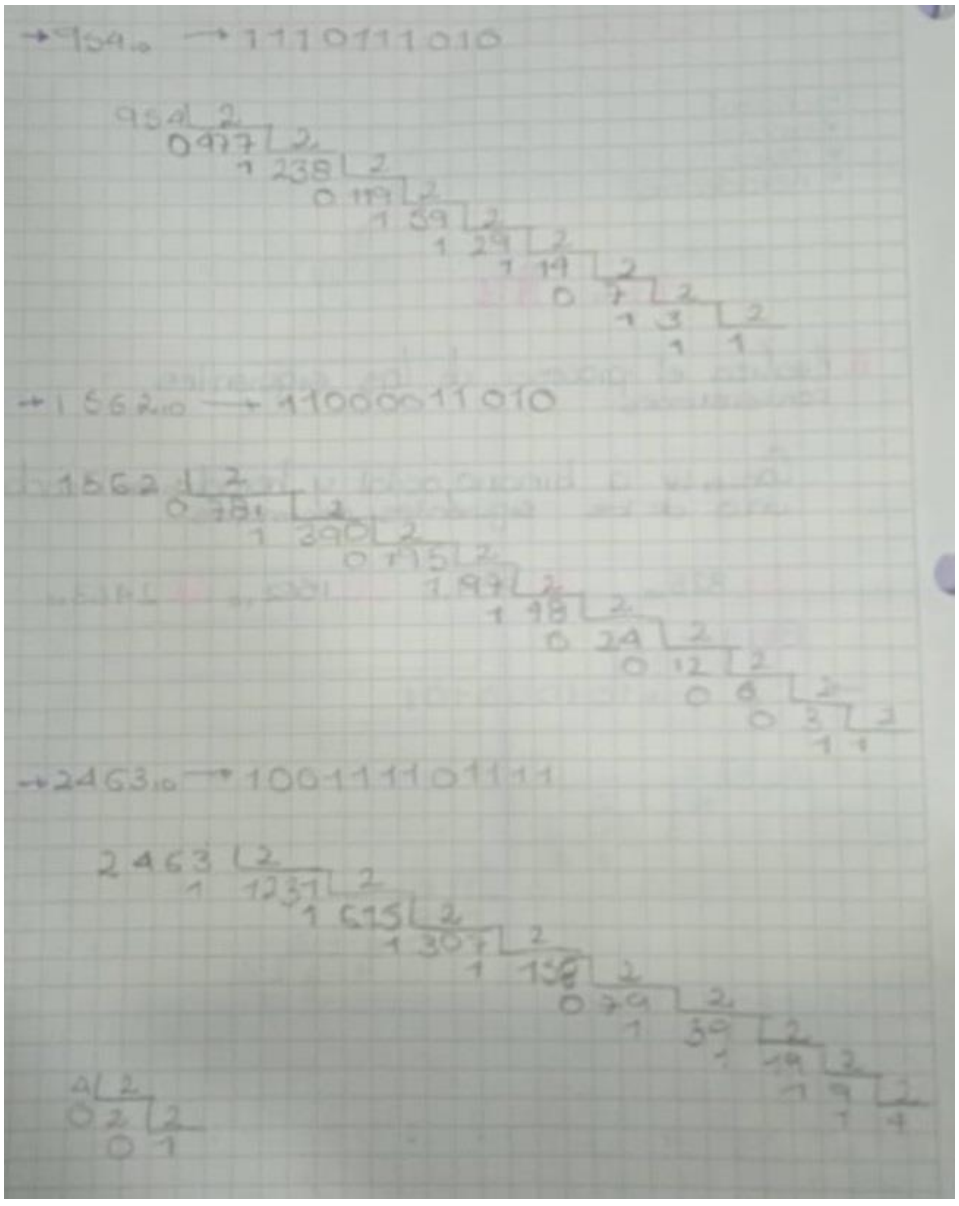
√a. 325_{10} √b 954_{10} √C 1562_{10} √d 2463_{10}

Binario

→ 325 ⟶ 1010001101

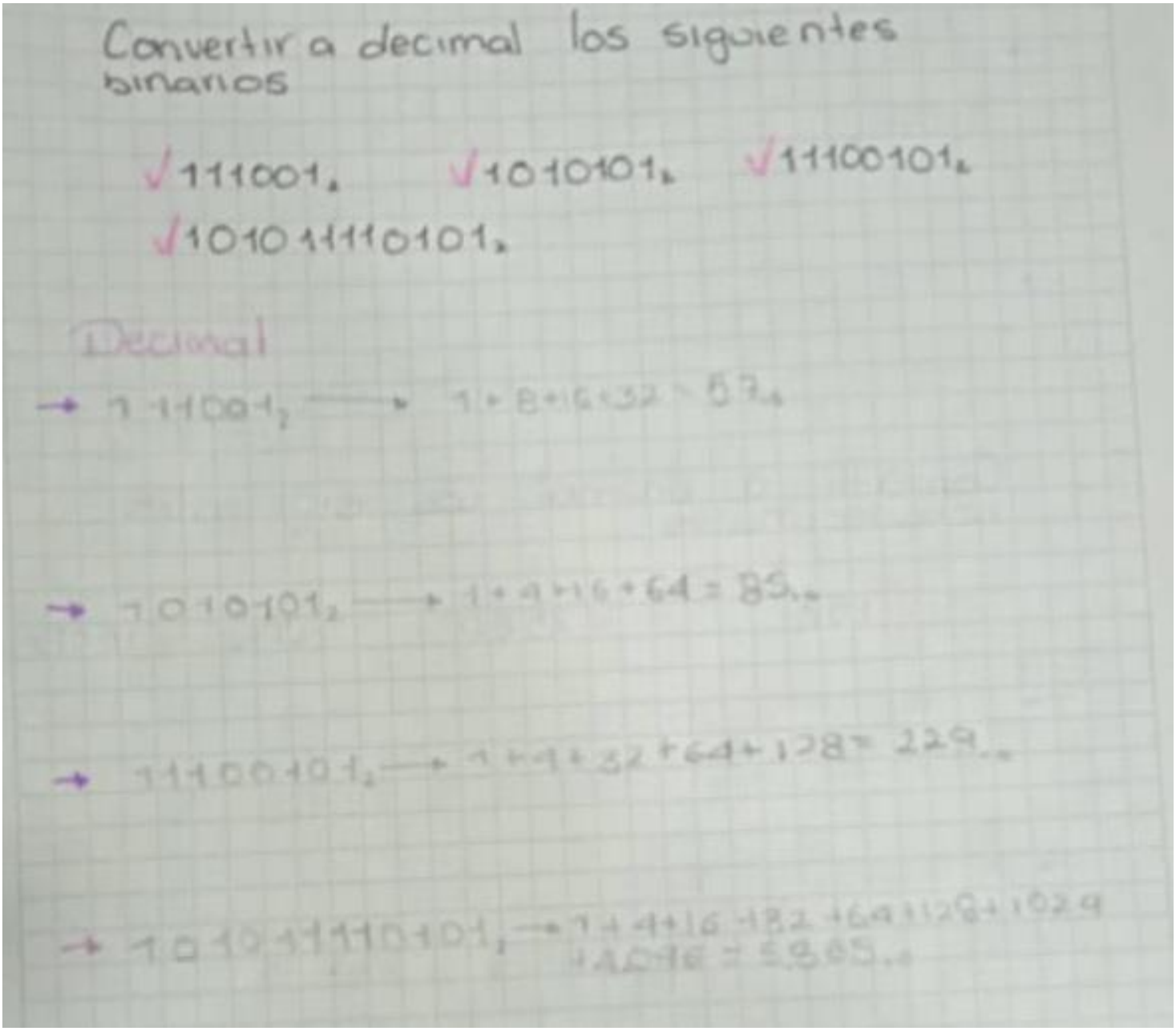
325 | 2
1 162 | 2
0 81 | 2
1 40 | 2
0 20 | 2
0 10 | 2
0 5 | 2
1 2 | 2
0 1

1010001101



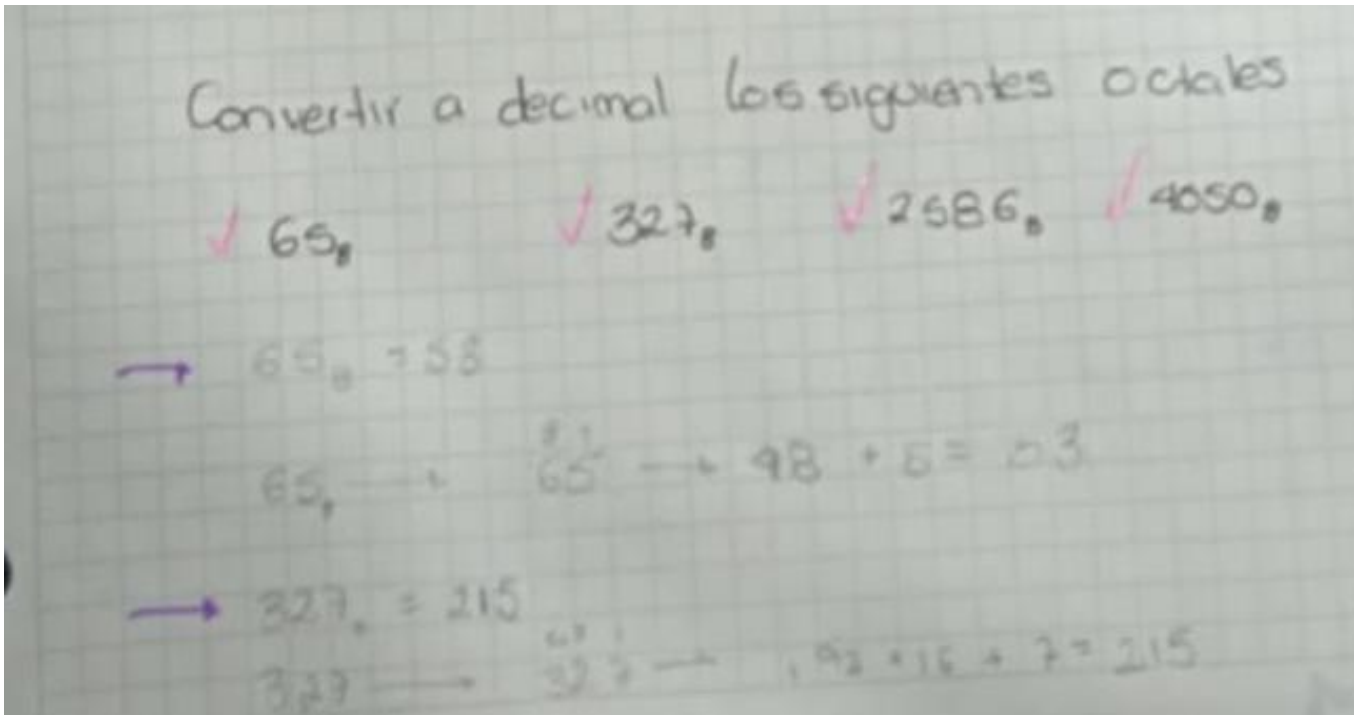
Convertir a decimal los siguientes binarios.

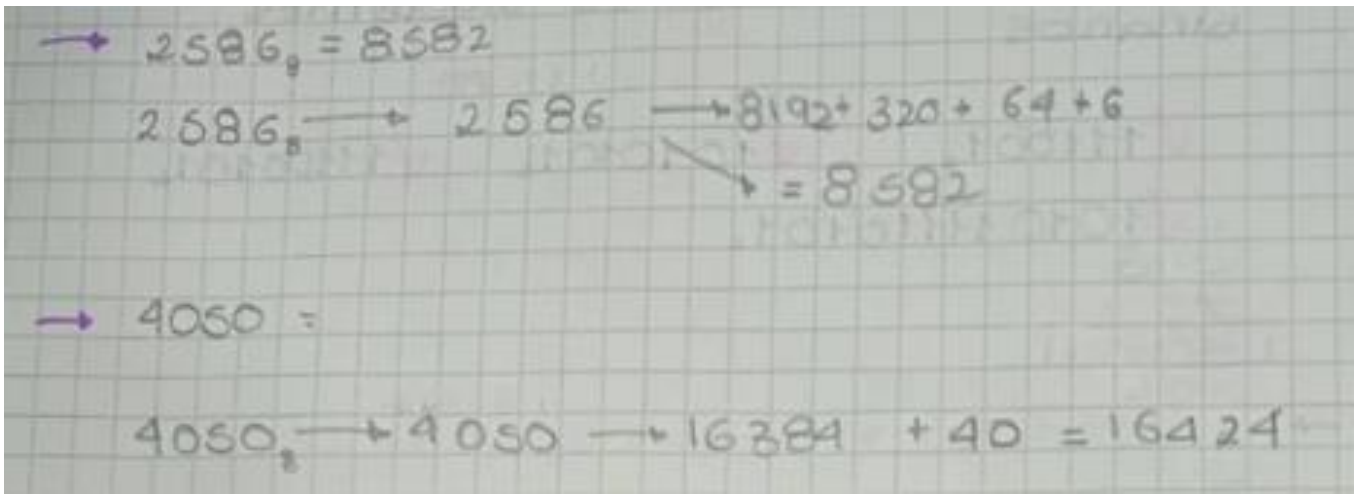
✓ a. 111001_2 b. 1010101_2 c. 11100101_2 d.101011110101_2



Convertir a decimal los siguientes octales.

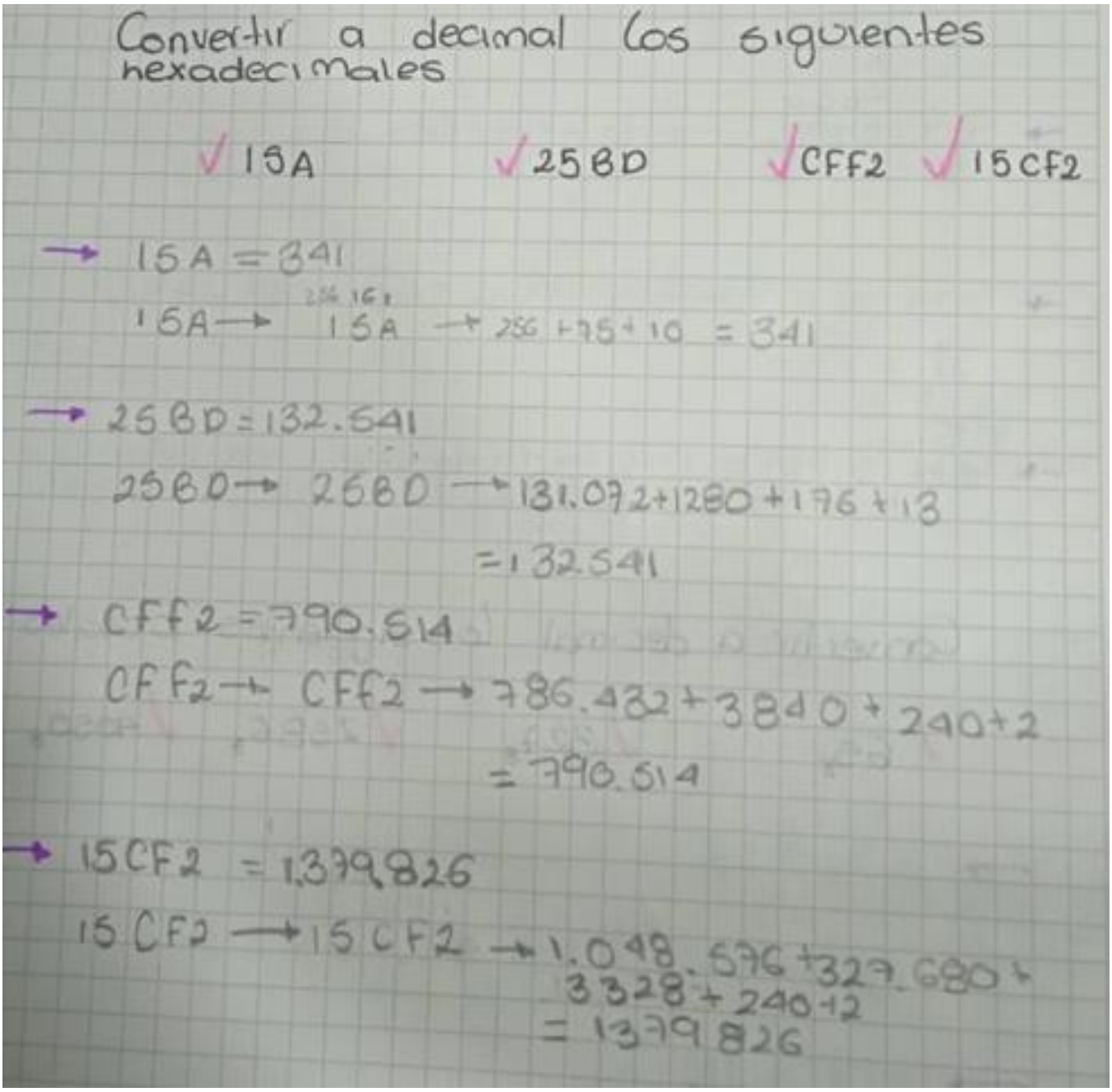
✓ a. 65_8 b. 327_8 c. 2586_8 d. 4050_8





Convertir a decimal los siguientes hexadecimales.

✓ a. $15A_{16}$ b. $25BD_{16}$ c. $CFF2_{16}$ d. $15CF2_{16}$

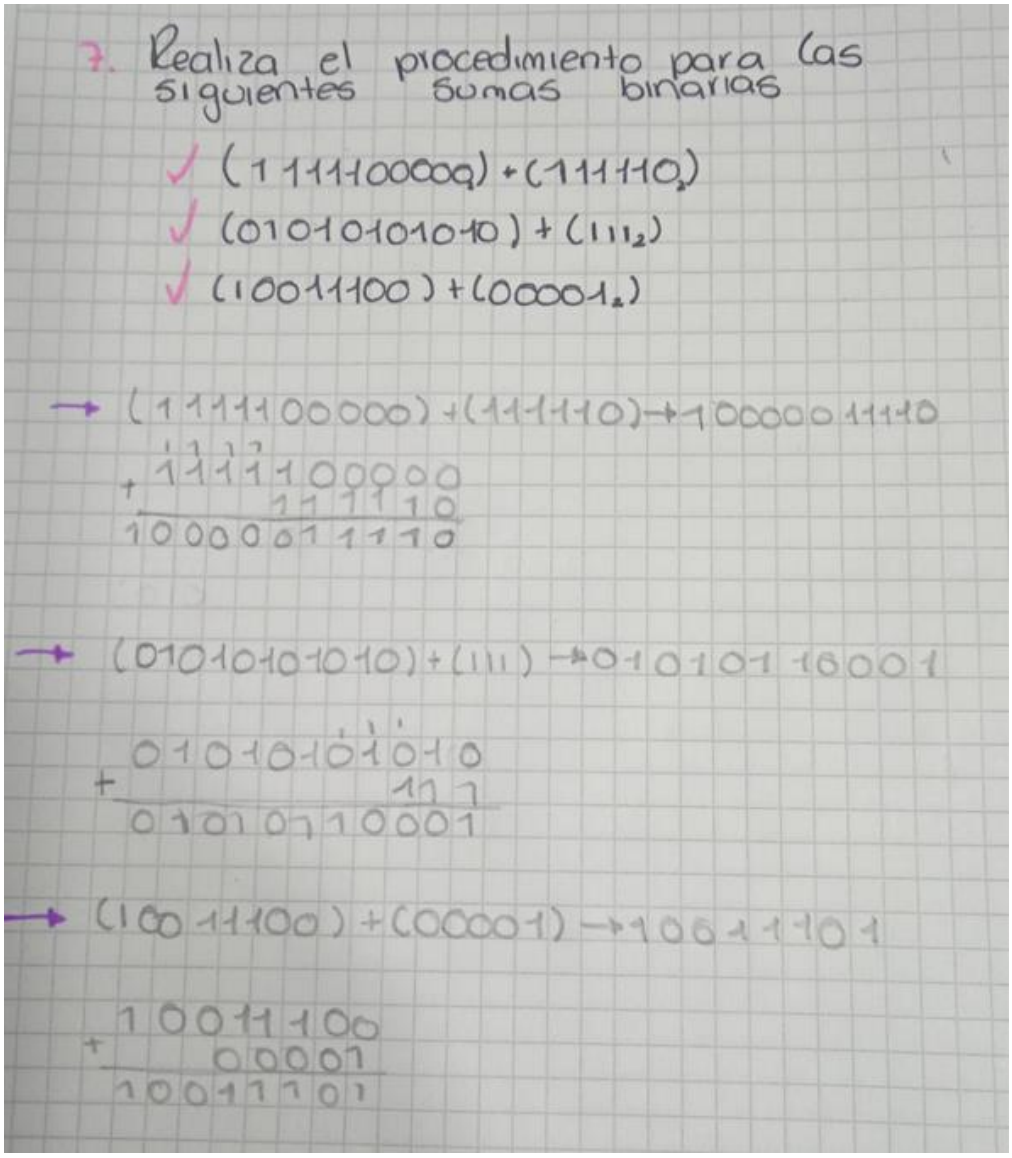


7. Realiza el procedimiento para las siguientes sumas binarias

✓ $(1111100000_2) + (111110_2)$

✓ $(01010101010_2) + (111_2)$

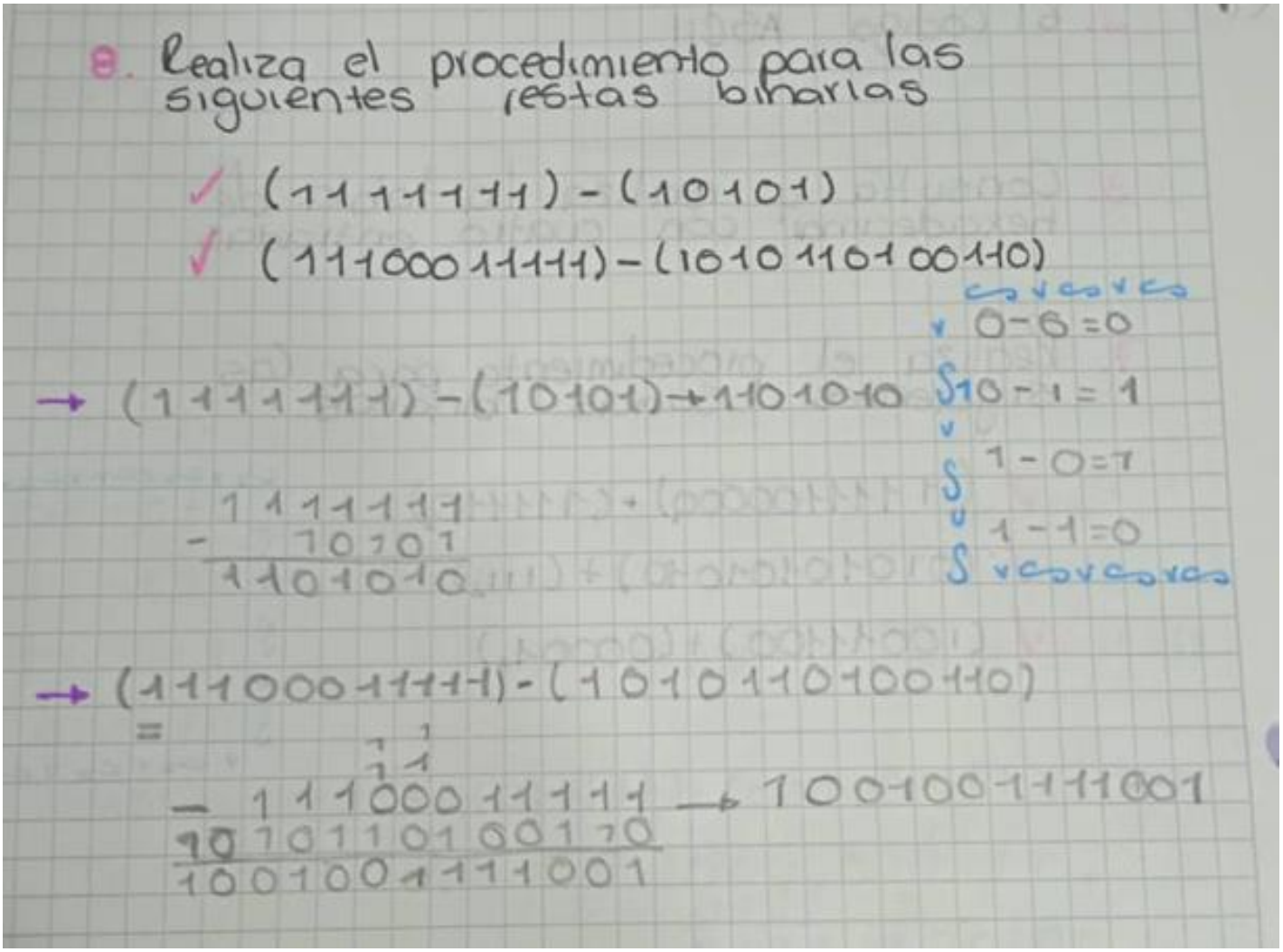
✓ $(10011100_2) + (00001_2)$

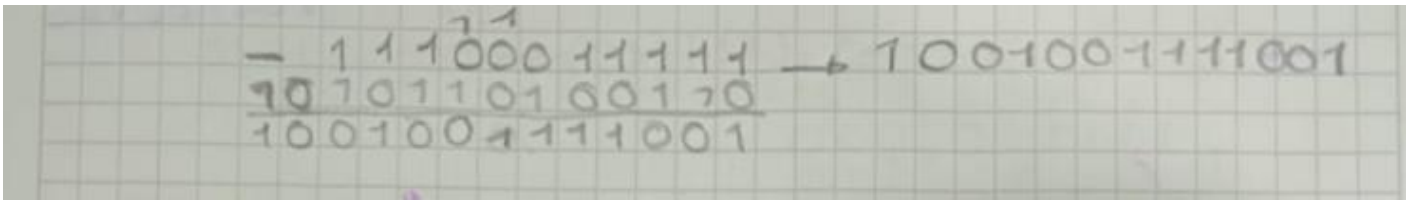


8. Realiza el procedimiento para las siguientes restas binarias

✓ $(1111111_2) - (10101_2)$

✓ $(11100011111_2) - (1010110100110_2)$

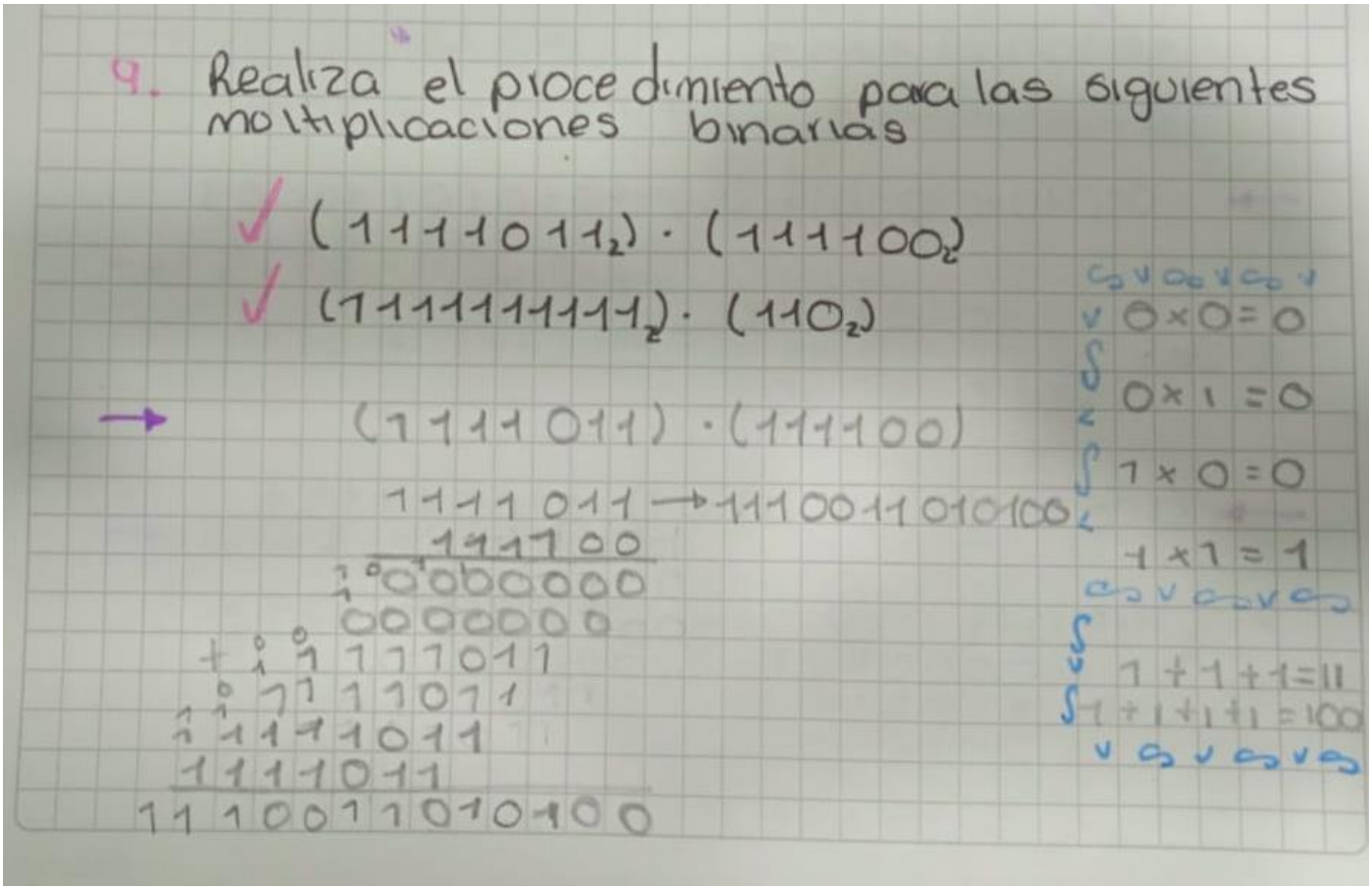


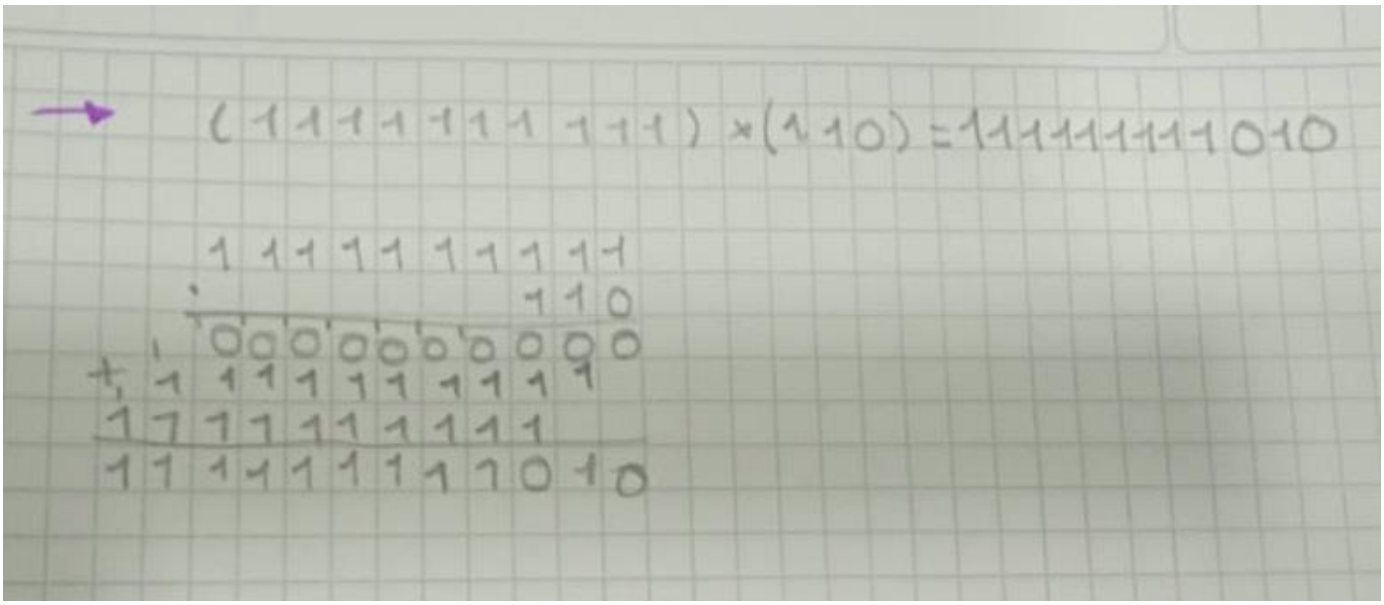


9. Realiza el procedimiento para las siguientes restas binarias

✓ $(1111011_2) * (111100_2)$

✓ $(1111111111_2) * (110_2)$





10. Describe la función de las teclas que se involucran al usar el código ASCII.

- **Teclas alfanuméricas (letras y números):** Estas teclas representan los caracteres alfabéticos y numéricos básicos. Cada letra y numero tiene un valor ASCII asociado. Por ejemplo, la tecla “A” tiene un valor ASCII de 65, la tecla "a" tiene un valor ASCII

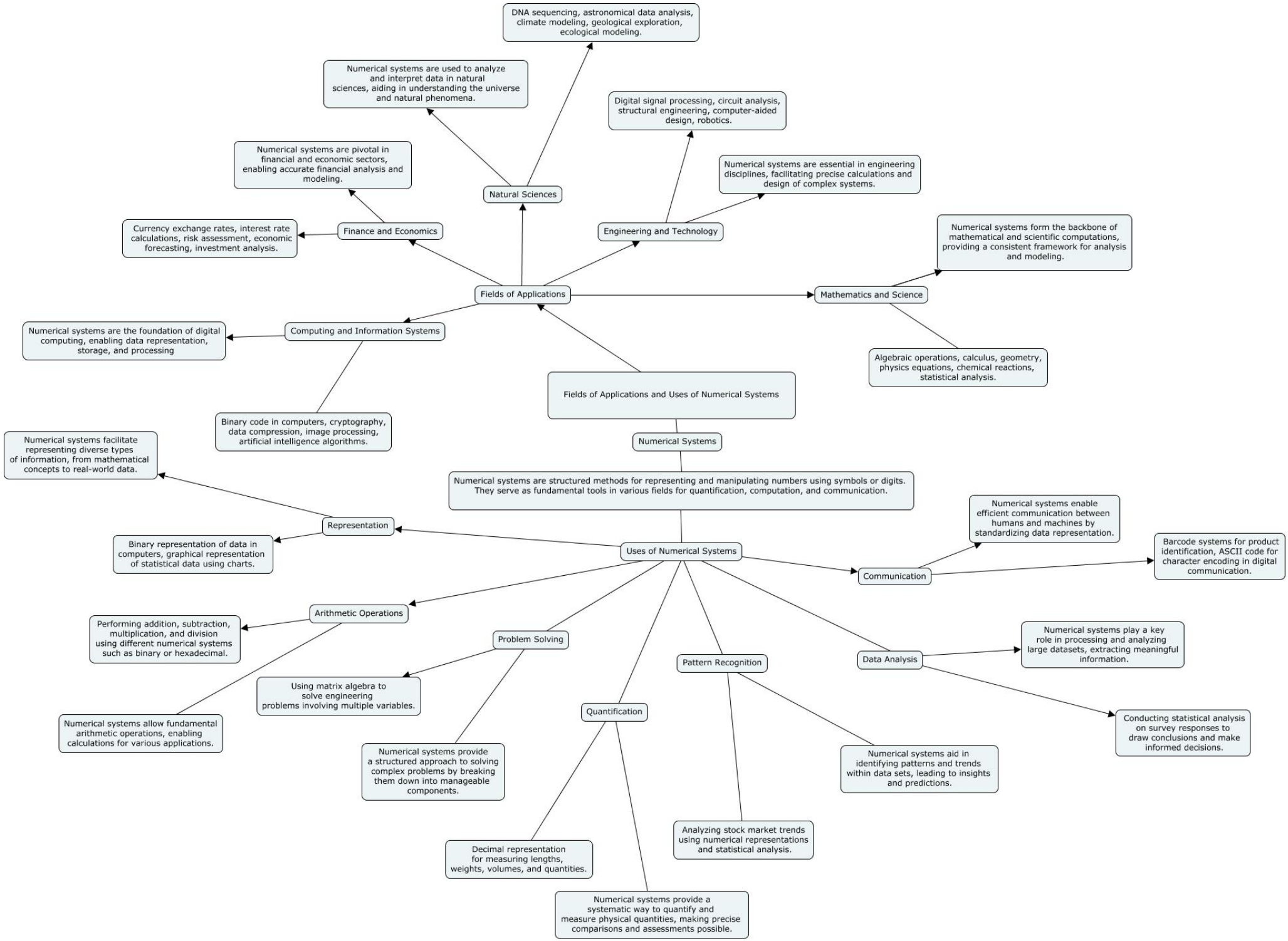
de 97 y la tecla "0" tiene un valor ASCII de 48. Para introducir estos caracteres, simplemente se presiona la tecla correspondiente.

- **Tecla de espacio en blanco:** La tecla de espacio en blanco introduce un espacio entre palabras o caracteres. En el código ASCII, el valor asociado al espacio en blanco es 32.
- **Teclas de puntuación y símbolos especiales:** Estas teclas incluyen signos de puntuación como comas, puntos y puntos y comas, así como símbolos especiales como el signo de exclamación (!), el símbolo de interrogación (?), entre otros. Cada uno de estos caracteres tiene un valor ASCII asignado, y puedes introducirlos presionando la tecla correspondiente.
- **Tecla "Shift":** La tecla "Shift" se utiliza para ingresar los caracteres alternativos en un teclado, como las letras mayúsculas y algunos símbolos. Al mantener presionada la tecla "Shift" mientras presionas una tecla alfanumérica, generalmente obtienes el carácter en mayúscula correspondiente. Por ejemplo, al presionar "Shift" + "A", obtendrás "A" en lugar de "a". Además, al mantener presionada la tecla "Shift" y presionar una tecla de puntuación, puedes ingresar los símbolos que están en la parte superior de las teclas alfanuméricas.
- **Teclas de control (Ctrl):** Las teclas "Ctrl" se utilizan junto con otras teclas para realizar acciones especiales en muchos programas y sistemas operativos. No tienen un valor ASCII directo, pero pueden combinarse con otras teclas para realizar comandos como "Ctrl + C" para copiar o "Ctrl + V" para pegar.
- **Tecla "Enter" o "Return":** Esta tecla se utiliza para confirmar una entrada, por ejemplo, al finalizar una línea de texto o al enviar un comando en un programa o terminal. En el código ASCII, la tecla "Enter" tiene un valor asociado, pero su uso no siempre implica la inserción del carácter correspondiente, ya que su función es principalmente de control.

11. Realice un algoritmo que permita pasar una numero de una base sea cual sea ”M” a otra base “N”, Nota M y N tienen como valor mínimo 1 y como valor máximo 16, además debe pasar el numero a binario y entregar El mensaje ASCII.

https://github.com/ARQUITECTURA-DE-HARDWARE-A1/Calculadora_de_bases/blob/main/code.python

ESQUEMA – PREGUNTAS ORIENTADORAS


Fields of Applications and Uses of Numerical Systems
Numerical Systems
Numerical systems are structured methods for representing and manipulating numbers using symbols or digits. They serve as fundamental tools in various fields for quantification, computation, and communication.
Fields of Applications
Natural Sciences
DNA sequencing, astronomical data analysis, climate modeling, geological exploration, ecological modeling.
Numerical systems are used to analyze and interpret data in natural sciences, aiding in understanding the universe and natural phenomena.
Engineering and Technology
Digital signal processing, circuit analysis, structural engineering, computer-aided design, robotics.
Numerical systems are essential in engineering disciplines, facilitating precise calculations and design of complex systems.
Finance and Economics
Numerical systems are pivotal in financial and economic sectors, enabling accurate financial analysis and modeling.
Currency exchange rates, interest rate calculations, risk assessment, economic forecasting, investment analysis.
Mathematics and Science
Numerical systems form the backbone of mathematical and scientific computations, providing a consistent framework for analysis and modeling.
Algebraic operations, calculus, geometry, physics equations, chemical reactions, statistical analysis.
Computing and Information Systems
Numerical systems are the foundation of digital computing, enabling data representation, storage, and processing
Binary code in computers, cryptography, data compression, image processing, artificial intelligence algorithms.
Uses of Numerical Systems
Representation
Numerical systems facilitate representing diverse types of information, from mathematical concepts to real-world data.
Binary representation of data in computers, graphical representation of statistical data using charts.
Communication
Numerical systems enable efficient communication between humans and machines by standardizing data representation.
Barcode systems for product identification, ASCII code for character encoding in digital communication.
Arithmetic Operations
Performing addition, subtraction, multiplication, and division using different numerical systems such as binary or hexadecimal.
Numerical systems allow fundamental arithmetic operations, enabling calculations for various applications.
Problem Solving
Using matrix algebra to solve engineering problems involving multiple variables.
Numerical systems provide a structured approach to solving complex problems by breaking them down into manageable components.
Quantification
Decimal representation for measuring lengths, weights, volumes, and quantities.
Numerical systems provide a systematic way to quantify and measure physical quantities, making precise comparisons and assessments possible.
Pattern Recognition
Analyzing stock market trends using numerical representations and statistical analysis.
Numerical systems aid in identifying patterns and trends within data sets, leading to insights and predictions.
Data Analysis
Numerical systems play a key role in processing and analyzing large datasets, extracting meaningful information.
Conducting statistical analysis on survey responses to draw conclusions and make informed decisions.