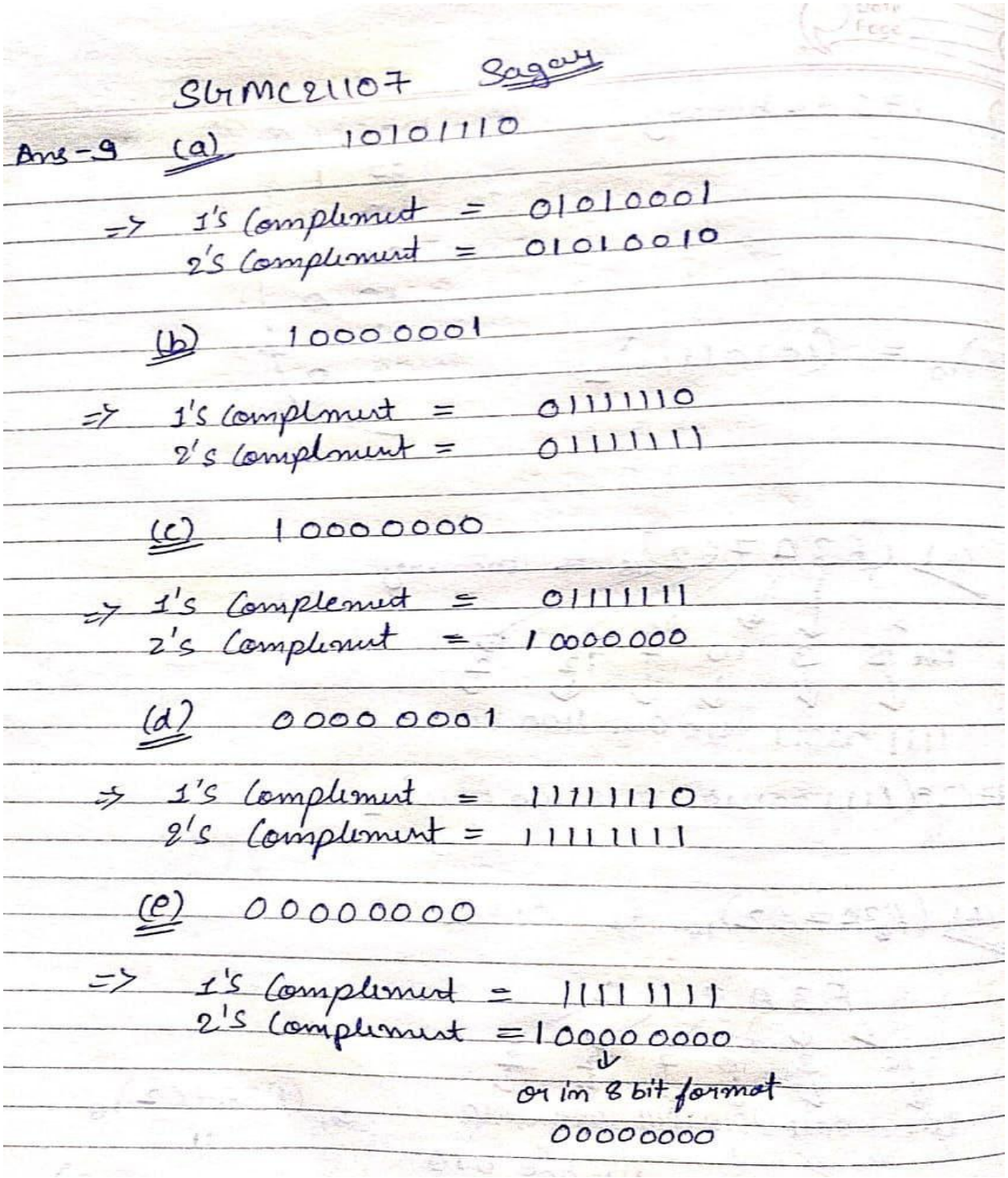
Q-9 Obtain the I's and 2's complements of the following eight-digit binary numbers:

a. **10101110**
b. **10000001**
c. **10000000**
d. **00000001**
e. **00000000**



SIrMC21107 Sagar

Ans-9 (a) 10101110

=> 1's Complement = 01010001
2's Complement = 01010010

(b) 10000001

=> 1's Complement = 01111110
2's Complement = 01111111

(c) 10000000

=> 1's Complement = 01111111
2's Complement = 10000000

(d) 00000001

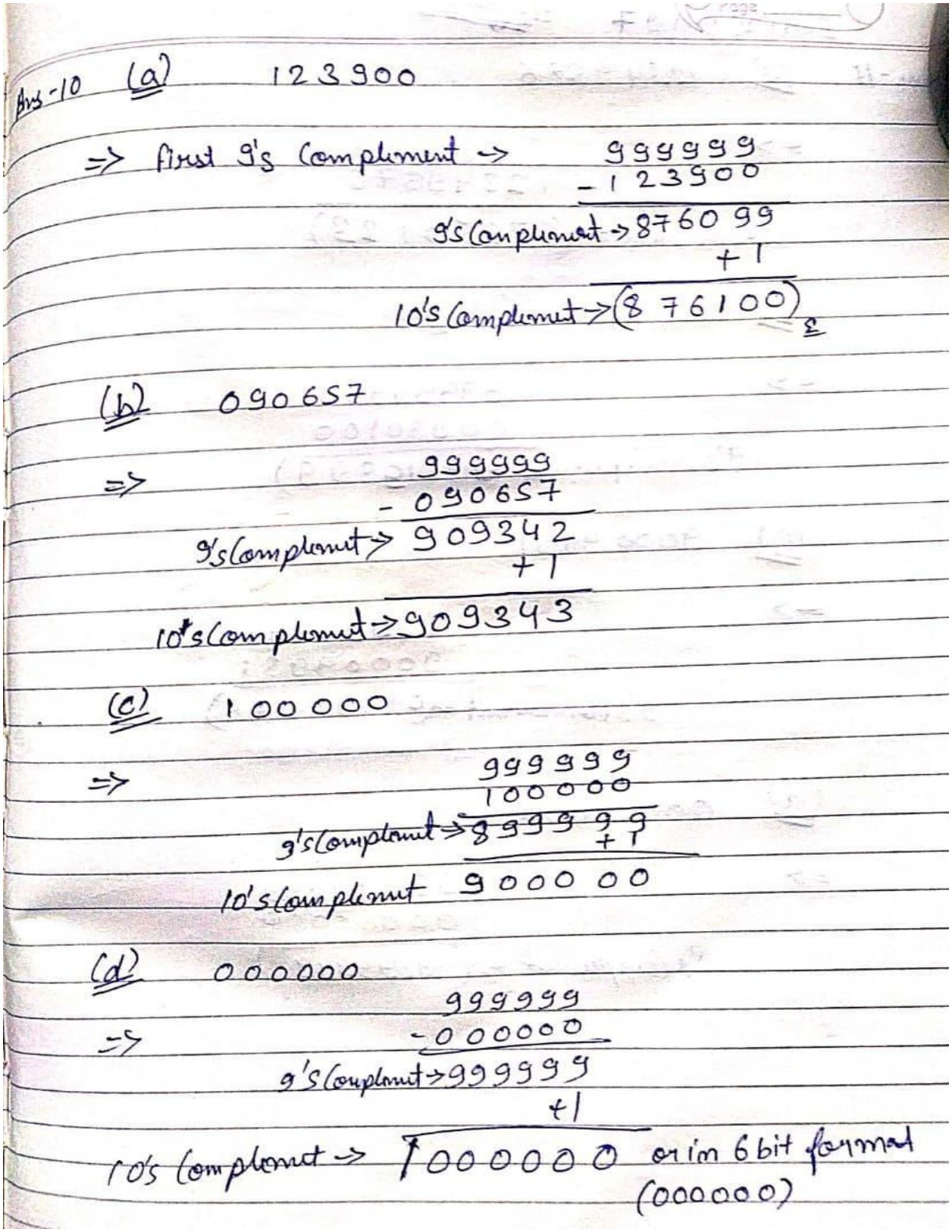
=> 1's Complement = 11111110
2's Complement = 11111111

(e) 00000000

=> 1's Complement = 11111111
2's Complement = 100000000
↓
or in 8 bit format
00000000

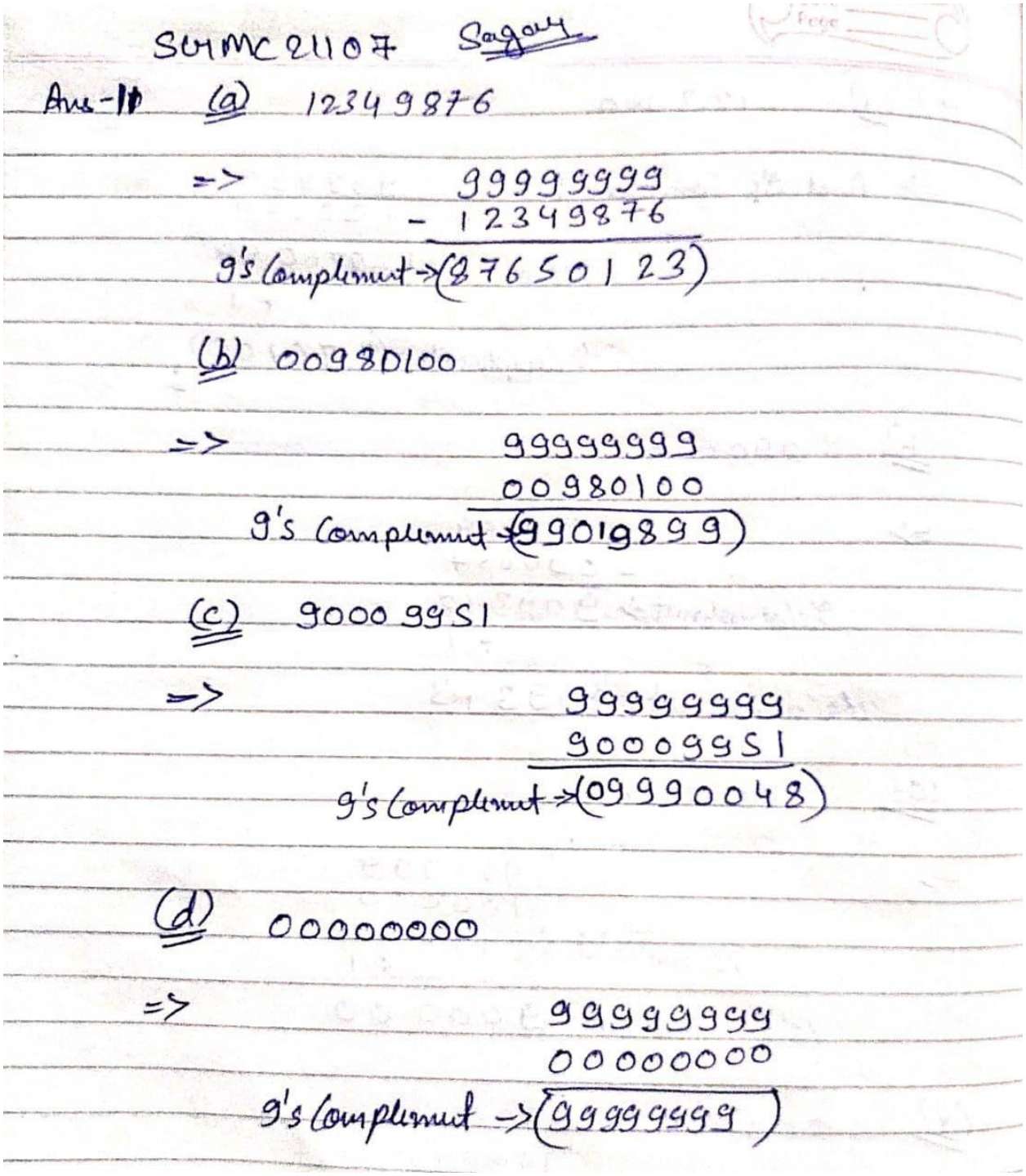
Q-10 Obtain the 10's complement of the following six-digit decimal numbers:

a. **123900**
b. **090657**
c. **100000**
d. **000000**



Q-11 Obtain the 9's complement of the following eight-digit decimal numbers:

a. **00980100**
b. **90009951**
c. **00000000**



Q-12 Perform the subtraction with the following unsigned binary numbers by taking the 2's complement of the subtrahend.

a. **11010 - 10000**
b. **11010 - 1101**
c. **100 - 110000**
d. **1010100 - 1010100**

Ans-12 (a) A B
11010 − 10000

=> 2's Complement of Subtrahend is

2's Complement of B = 1's Complement + 1
= 01111 + 1
= 10000

Now,
A 11010
+B 10000
101010

dropping Carry 1

So,
(11010 − 10000) = (01010)

(b) A B
11010 − 01101

2's Complement of B = 10010 + 1
= 10011

Now,
A 11010
+B 01101
10111

A 11010
+B 10011
101101

dropping Carry 1

So,
11010 − 1101 = 01101

(c) $\overset{A}{100} - \overset{B}{110000}$

$\Rightarrow$ 2's Complement of B = 001111 + 1
= 010000

Now, A 000100
+ B 010000
010100

there is no carry so we will find 2's Complement of Number we got

2's Complement = 101011 + 1
= 101100

as there was no carry before So, No. we got is negative

So, (100 − 110000) = (−101100)

(d) $10101\overset{A}{00} - 10\overset{B}{1}0100$

$\Rightarrow$ 2's Complement of B = 0101011 + 1
= 0101100

Now A 1010100
+ B 0101100
1 0000000

dropping carry

So, (1010100 − 1010100) = (0000000)

Q-13 Perform the arithmetic operations (+42) + (- 13) and (-42) - (-13) in binary using signed-2's complement representation for negative numbers.

BCA/MC21107 Sagar

Ans-13 (a) $(+42) + (-13)$

$\Rightarrow +42 = (101010)_2$, $-13 = (-1101)_2$

$101010 + (-1101)$

2's complement of B = 110010 + 1

= 110011

Now,

$$\begin{array}{rr} & A\ 101010 \\ +B & 110011 \\ \hline & 1011101 \end{array}$$

dropping carry 1

So, $(011101)_2$

$\Rightarrow 0+16+8+4+0+1$

$\Rightarrow (29)_{10}$

(b) $(-42)-(-13)$

$\Rightarrow -42 + (-(-13))$

$\Rightarrow -42 + 13$

$\Rightarrow 13 - 42$

$13 = (1101)_2$, $-42 = (-101010)_2$

2's complement of B = 010101 + 1

= 010110

Now A 001101

+B 010110

100011

No Carry So we find the 2's Complemd of answer.

2's Complemt = 011100+1

= 011101

It will be -ve

So, $(-011101)_2$

$= (-29)_{10}$ ≦

Q-14 (Q-14 Was Not in Assignment pdf)

Q-15 Perform the arithmetic operations (+70) + (+80) and (-70) + (-80) with binary numbers in signed-2's complement representation. Use eight bits to accommodate each number together with its sign. Show that overflow occurs in both cases, that the last two carries are unequal, and that there is a sign reversal.

SGMC21107 Sagar Gupta Sagar

Ans-15

(a) $\overset{A}{(+70)} + \overset{B}{(+80)}$

$\Rightarrow$ 70 = 01000110 , 80 = 01010000

As both +ve So,

$$\begin{array}{rl} A & 01000110 \\ +B & 01010000 \\ \hline & 10010110 \quad (-ve) \end{array}$$

As there is no extra carry means it is in 8 bit limit So, Sign will change $-(10010110)$, Hence overflow occurs

(b) $\overset{A}{(-70)} + \overset{B}{(-80)}$

$\Rightarrow$ As both A and B are -ve & So we will take

2's Complement of A = 10111001 + 1 = 10111010

2's Complemet of B = 10101111 + 1 = 10110000

Than,

$$\begin{array}{rl} A & 10111010 \\ +B & 10110000 \\ \hline & 101101010 \quad (+ve) \end{array}$$

As there is 1 Carry So, it will be (+ve)
Hence Sum of 2 (-ve) number is (+ve)
overflow occurs

Q-16 Represent decimal number 8620 in

(a) BCD

(b) excess-3 code

(c) 2421 code

(d) as a binary number

Ans-16 (a) $(8620)_{10}$ in BCD

=> 8 6 2 0

1000 0110 0010 0000

So, (8620) = 1000 0110 0010 0000

(b) $(8620)_{10}$ in excess-3 Code

=> 8 6 2 0

+3 +3 +3 +3

11 9 5 3

1011 1001 0101 0011

So, $(8620)_{10}$ = 1011 1001 0101 0011

(c) $(8620)_{10}$ in 2421 - Code

=> 8 6 2 0

1110 1100 0010 0000

So, $(8620)_{10}$ = 1110 1100 0010 0000

(d) 8620 in binary number

=>

2 | 8620	0
4310	0
2155	1
1077	1
538	0
269	1
134	0
67	1
33	1
16	0
8	0
4	0
2	0
1	

So, $(8620)_{10} = (10000110101100)_2$