



SYMBIOSIS INSTITUTE OF TECHNOLOGY, NAGPUR

Symbiosis International (Deemed University)

(Established under section 3 of the UGC Act, 1956)

Re-accredited by NAAC with 'A+' Grade | Awarded Category - I by UGC

Founder: Prof. Dr. S. B. Mujumdar, M. Sc., Ph. D. (Awarded Padma Bhushan and Padma Shri by President of India)

MES CA-1

NAME: Anaya Shende

PRN: 24070521213

SECTION: D

SEMESTER: 4th

Q.1. Write an 8051 Assembly Language Program (ALP) to generate the last four digits of your PRN using arithmetic instructions only. The final result should be available in the Accumulator (A).

The screenshot shows the Proteus SIM software interface. On the left, the '8051' hardware component is displayed with its pins labeled. The assembly code window shows the following program:

```
RST|Step|Pause|New|Load|Save|CPI|Paste|BP| Time: 3ms 227us - Instructions: 3226
ORG 0000H
0000| MOV A, #06H ; A = 6
0002| MOV B, #06H ; B = 6
0005| ADD A, B ; A = 12 (BCD)
0007| DA A ; Decimal adjust
0008| MOV B, A ; Store 12 in B
000A| INC A ; A = 13
END

P0.7 1 Display-select Decoder CS|DAC WR
P0.6 1 Keypad Column 2
P0.5 1 Keypad Column 1
P0.4 1 Keypad Column 0
P0.3 1 Keypad Row 3
P0.2 1 Keypad Row 2
P0.1 1 Keypad Row 1
P0.0 1 Keypad Row 0
P1.7 1 LED 7|Seg. dp|DAC DB7|LCD DB7
P1.6 1 LED 6|Seg. g|DAC DB6|LCD DB6
P1.5 1 LED 5|Seg. f|DAC DB5|LCD DB5
P1.4 1 LED 4|Seg. e|DAC DB4|LCD DB4
P1.3 1 LED 3|... d|..DB3|..DB3|.. RS
P1.2 1 LED 2|... c|..DB2|..DB2|LCD E
P1.1 1 LED 1|Seg. b|DAC DB1|LCD DB1
P1.0 1 LED 0|Seg. a|DAC DB0|LCD DB0
P2.7 1 SW 7|ADC DB7
P2.6 1 SW 6|ADC DB6
P2.5 1 SW 5|ADC DB5
P2.4 1 SW 4|ADC DB4
P2.3 1 SW 3|ADC DB3
P2.2 1 SW 2|ADC DB2
P2.1 1 SW 1|ADC DB1
P2.0 1 SW 0|ADC DB0
P3.7 1 ADC RD|Comparator Output
P3.6 1 ADC WR
P3.5 1 Motor Sensor
P3.4 1 Display-select Input 1
P3.3 1 AND Gate Output|Display-se...t 0
P3.2 1 ADC INTR
P3.1 1 Motor Control Bit 1|Ext. UART Rx
P3.0 1 Motor Control Bit 0|Ext. UART Tx
```

The bottom half of the screen shows the hardware setup. It includes a digital input section with switches for DI, LD, and various keybounce settings. There is also a DAC output section connected to a scope. On the right, there is a 4-digit LED display showing '8888'. Below the display are several control buttons and indicators, including 'Rx Reset', 'Tx Send', 'MAX', 'MIN', and 'Motor Enabled'. A color calibration target is also present.

Q.2. Write an 8051 Assembly Language Program to compare two unsigned numbers stored at internal RAM locations 50H and 51H without using CJNE, DJNZ, or SUBB. Only MOV, INC, DEC, JZ, JNZ, CLR, SETB, ANL, and ORL instructions are allowed. Store the result such that 01H indicates $50H > 51H$, 00H indicates equality, and FFH indicates $50H < 51H$.

EDSIM5 J1

System Clock (MHz) 12.0 1 Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x00	0x00	0x00	R6	0x00	ACC	0x00
RXD	TXD	1	1	R5	0x00	PSW	0x00
SCON	0x00	TCON	0x00	R4	0x00	IP	0x00
pins	bits	TH1	TL1	R3	0x00	IE	0x00
0xFF	0xFF P3	0x00	0x00	R2	0x00	PCON	0x00
0xFF	0xFF P2	PC		R1	0x00	DPH	0x00
0xFF	0xFF P1			R0	0x02	DPL	0x00
0xFF	0xFF P0					SP	0x07

8051

Modify RAM
Data Memory
addr 0x00 0x05 value

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	06	04	01	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Copyright ©2005-2024 James Rogers Remove All Breakpoints

Time: 196us - Instructions: 178

```

0008| MOV 50H, #06H
0009| MOV 51H, #04H
000A| MOV A, 50H
000B| MOV R0, A ; store first number
000C| MOV A, 51H
000D| MOV R1, A ; store second number

START:
000E| MOV A, R0
000F| JZ CHECK_R1 ; if first becomes zero
0010| JZ FIRST_BIG ; if second becomes zero

0011| DEC R0
0012| DEC R1
0013| SJMP START

CHECK_R1:
0014| SJMP END

FIRST_BIG:
0015| MOV A, R1
0016| JZ SAME
0017| MOV 52H, #0FFH ; first < second
0018| SJMP END

SAME:
0019| MOV 52H, #01H ; first > second
0020| SJMP END

END:
0021| SJMP END

```

P0.7 1 Display-select Decoder CS|DAC WR
P0.6 1 Keypad Column 2
P0.5 1 Keypad Column 1
P0.4 1 Keypad Column 0
P0.3 1 Keypad Row 3
P0.2 1 Keypad Row 2
P0.1 1 Keypad Row 1
P0.0 1 Keypad Row 0
P1.7 1 LED 7|Seg. dp|DAC DB7|LCD DB7
P1.6 1 LED 6|Seg. g|DAC DB6|LCD DB6
P1.5 1 LED 5|Seg. f|DAC DB5|LCD DB5
P1.4 1 LED 4|Seg. e|DAC DB4|LCD DB4
P1.3 1 LED 3|... d|..DB3|..DB3|... RS
P1.2 1 LED 2|... c|..DB2|..DB2|LCD E
P1.1 1 LED 1|Seg. b|DAC DB1|LCD DB1
P1.0 1 LED 0|Seg. a|DAC DB0|LCD DB0
P2.7 1 SW 7|ADC DB7
P2.6 1 SW 6|ADC DB6
P2.5 1 SW 5|ADC DB5
P2.4 1 SW 4|ADC DB4
P2.3 1 SW 3|ADC DB3
P2.2 1 SW 2|ADC DB2
P2.1 1 SW 1|ADC DB1
P2.0 1 SW 0|ADC DB0
P3.7 1 ADC RD|Comparator Output
P3.6 1 ADC WR
P3.5 1 Motor Sensor
P3.4 1 Display-select Input 1
P3.3 1 AND Gate Output|Display-select 0
P3.2 1 ADC INTR
P3.1 1 Motor Control Bit 1|Ext. UART Rx
P3.0 1 Motor Control Bit 0|Ext. UART Tx

EDSIM5 J1

System Clock (MHz) 12.0 1 Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x00	0x00	0x00	R6	0x00	ACC	0x00
RXD	TXD	1	1	R5	0x00	PSW	0x00
SCON	0x00	TCON	0x00	R4	0x00	IP	0x00
pins	bits	TH1	TL1	R3	0x00	IE	0x00
0xFF	0xFF P3	0x00	0x00	R2	0x00	PCON	0x00
0xFF	0xFF P2	PC		R1	0x00	DPH	0x00
0xFF	0xFF P1			R0	0x02	DPL	0x00
0xFF	0xFF P0					SP	0x07

8051

Modify RAM
Data Memory
addr 0x00 0x05 value

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	06	04	01	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Copyright ©2005-2024 James Rogers Remove All Breakpoints

Time: 395us - Instructions: 377

```

0012| DEC R0
0013| DEC R1
0014| SJMP START

CHECK_R1:
0015| MOV A, R1
0016| JZ SAME
0017| MOV 52H, #0FFH ; first < second
0018| SJMP END

FIRST_BIG:
0019| MOV 52H, #01H ; first > second
0020| SJMP END

SAME:
0021| MOV 52H, #00H ; equal
0022| SJMP END

END:
0023| SJMP END

```

P0.7 1 Display-select Decoder CS|DAC WR
P0.6 1 Keypad Column 2
P0.5 1 Keypad Column 1
P0.4 1 Keypad Column 0
P0.3 1 Keypad Row 3
P0.2 1 Keypad Row 2
P0.1 1 Keypad Row 1
P0.0 1 Keypad Row 0
P1.7 1 LED 7|Seg. dp|DAC DB7|LCD DB7
P1.6 1 LED 6|Seg. g|DAC DB6|LCD DB6
P1.5 1 LED 5|Seg. f|DAC DB5|LCD DB5
P1.4 1 LED 4|Seg. e|DAC DB4|LCD DB4
P1.3 1 LED 3|... d|..DB3|..DB3|... RS
P1.2 1 LED 2|... c|..DB2|..DB2|LCD E
P1.1 1 LED 1|Seg. b|DAC DB1|LCD DB1
P1.0 1 LED 0|Seg. a|DAC DB0|LCD DB0
P2.7 1 SW 7|ADC DB7
P2.6 1 SW 6|ADC DB6
P2.5 1 SW 5|ADC DB5
P2.4 1 SW 4|ADC DB4
P2.3 1 SW 3|ADC DB3
P2.2 1 SW 2|ADC DB2
P2.1 1 SW 1|ADC DB1
P2.0 1 SW 0|ADC DB0
P3.7 1 ADC RD|Comparator Output
P3.6 1 ADC WR
P3.5 1 Motor Sensor
P3.4 1 Display-select Input 1
P3.3 1 AND Gate Output|Display-select 0
P3.2 1 ADC INTR
P3.1 1 Motor Control Bit 1|Ext. UART Rx
P3.0 1 Motor Control Bit 0|Ext. UART Tx

EDSIM5 J1

System Clock (MHz) 12.0 1 Update Freq.

SBUF

R/O	W/O	TH0	TL0	R7	0x00	B	0x00
0x00	0x00	0x00	0x00	R6	0x00	ACC	0x00
RXD	TXD	1	1	R5	0x00	PSW	0x00
SCON	0x00	TCON	0x00	R4	0x00	IP	0x00
pins	bits	TH1	TL1	R3	0x00	IE	0x00
0xFF	0xFF P3	0x00	0x00	R2	0x00	PCON	0x00
0xFF	0xFF P2	PC		R1	0x00	DPH	0x00
0xFF	0xFF P1			R0	0x02	DPL	0x00
0xFF	0xFF P0					SP	0x07

8051

Modify RAM
Data Memory
addr 0x00 0x05 value

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	06	04	01	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Copyright ©2005-2024 James Rogers Remove All Breakpoints

Time: 395us - Instructions: 377

```

0012| DEC R0
0013| DEC R1
0014| SJMP START

CHECK_R1:
0015| MOV A, R1
0016| JZ SAME
0017| MOV 52H, #0FFH ; first < second
0018| SJMP END

FIRST_BIG:
0019| MOV 52H, #01H ; first > second
0020| SJMP END

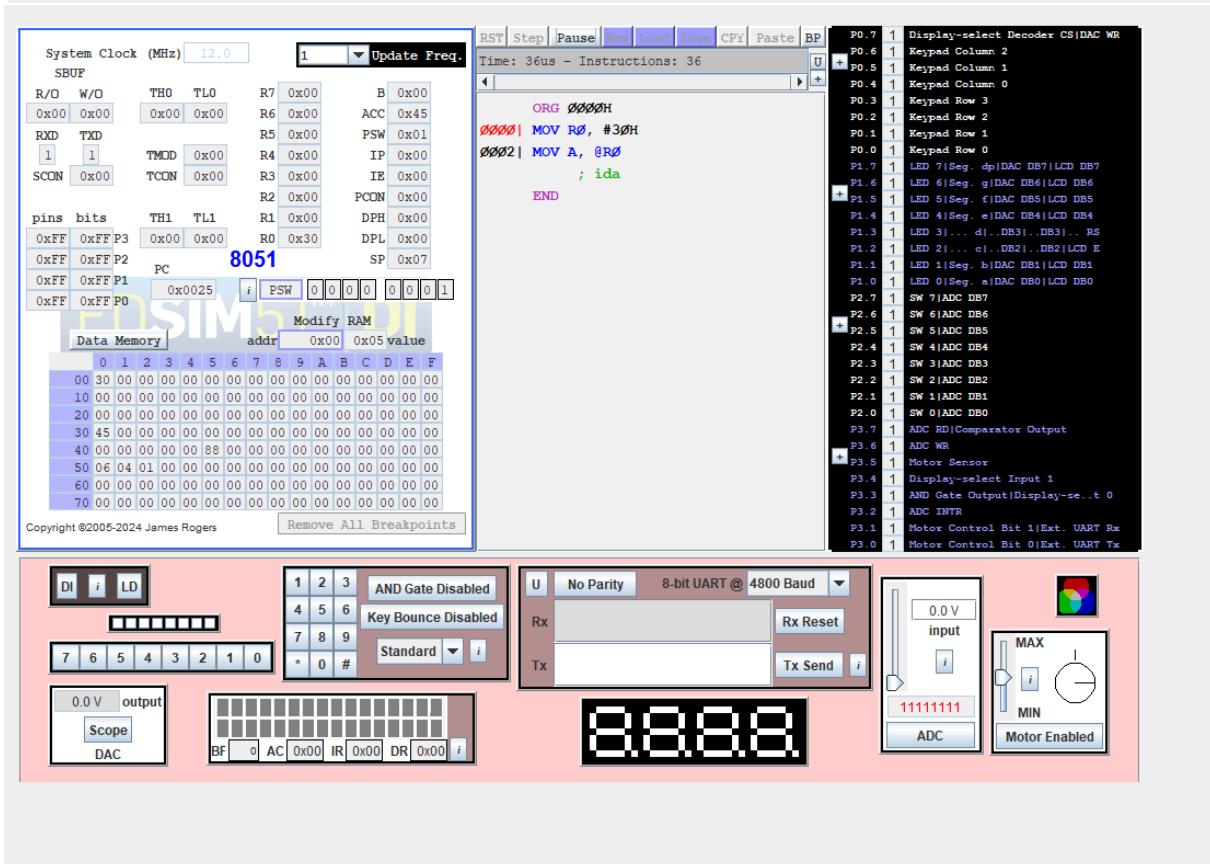
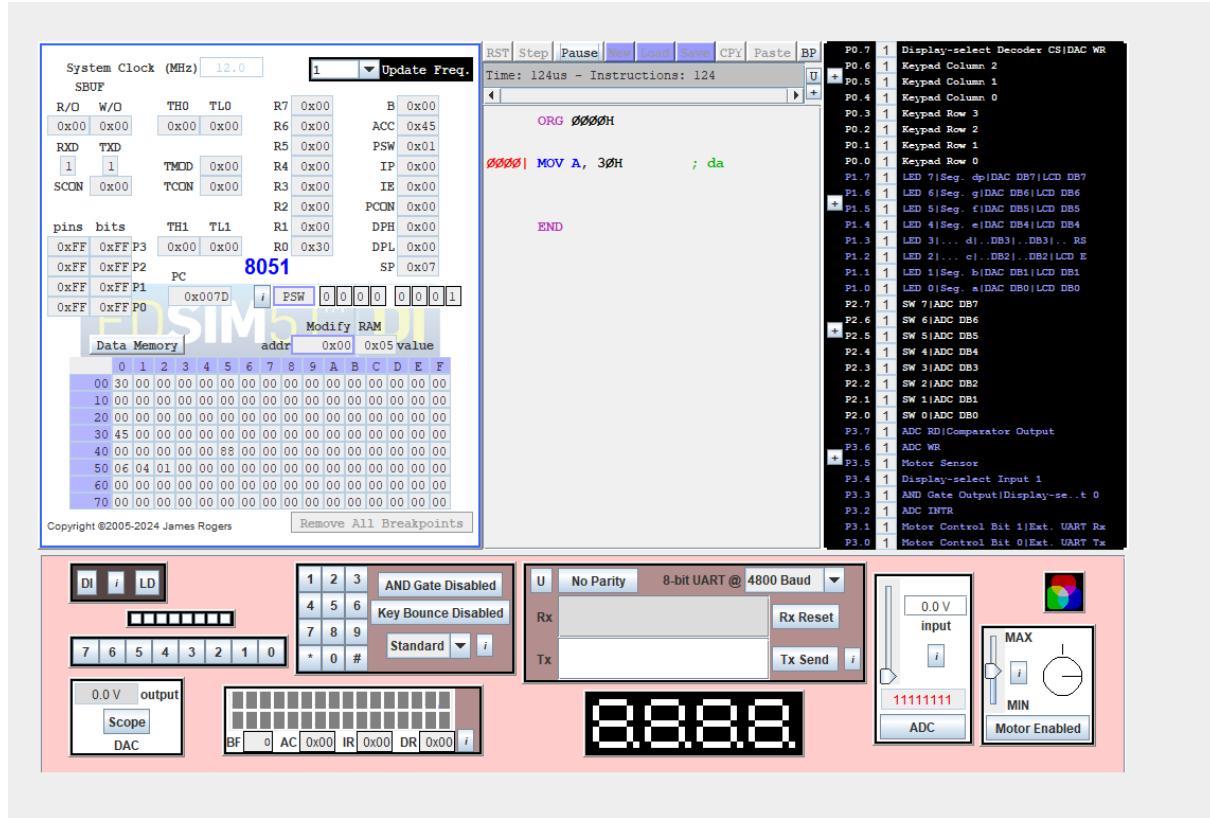
SAME:
0021| MOV 52H, #00H ; equal
0022| SJMP END

END:
0023| SJMP END

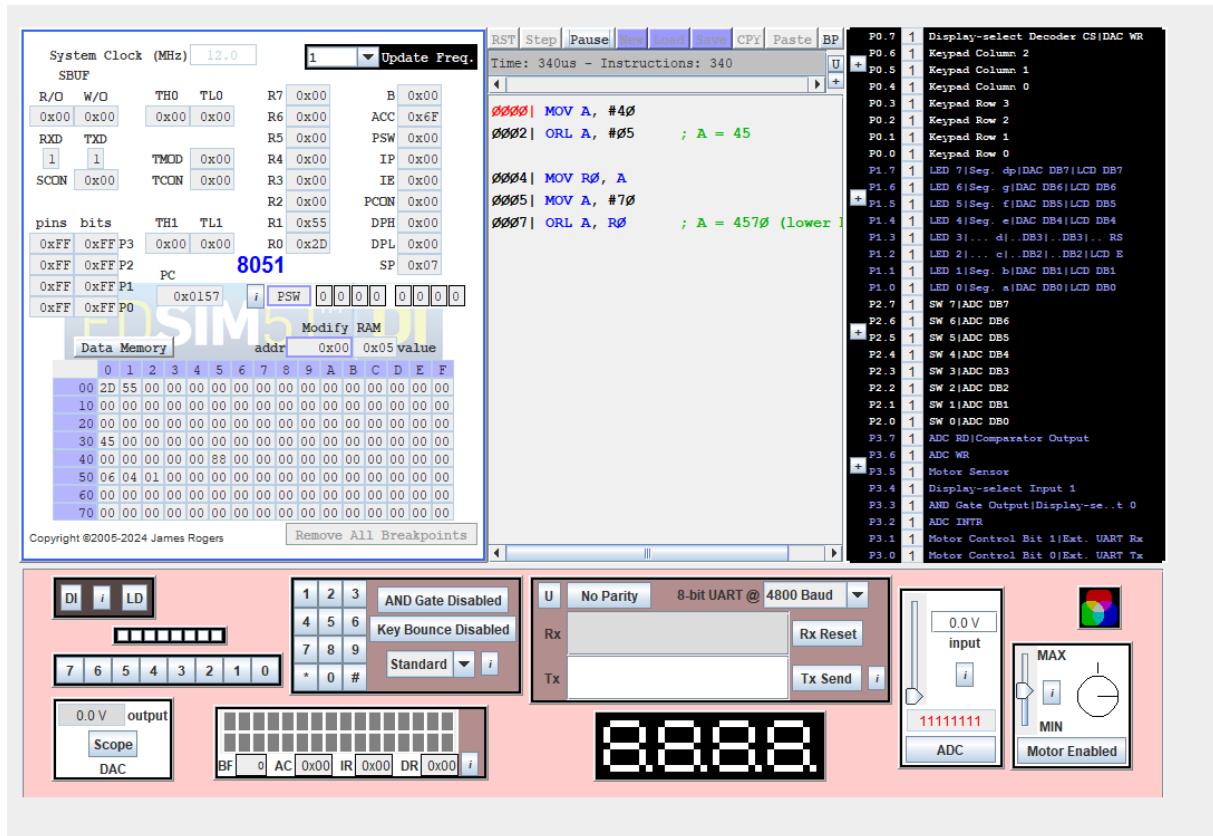
```

P0.7 1 Display-select Decoder CS|DAC WR
P0.6 1 Keypad Column 2
P0.5 1 Keypad Column 1
P0.4 1 Keypad Column 0
P0.3 1 Keypad Row 3
P0.2 1 Keypad Row 2
P0.1 1 Keypad Row 1
P0.0 1 Keypad Row 0
P1.7 1 LED 7|Seg. dp|DAC DB7|LCD DB7
P1.6 1 LED 6|Seg. g|DAC DB6|LCD DB6
P1.5 1 LED 5|Seg. f|DAC DB5|LCD DB5
P1.4 1 LED 4|Seg. e|DAC DB4|LCD DB4
P1.3 1 LED 3|... d|..DB3|..DB3|... RS
P1.2 1 LED 2|... c|..DB2|..DB2|LCD E
P1.1 1 LED 1|Seg. b|DAC DB1|LCD DB1
P1.0 1 LED 0|Seg. a|DAC DB0|LCD DB0
P2.7 1 SW 7|ADC DB7
P2.6 1 SW 6|ADC DB6
P2.5 1 SW 5|ADC DB5
P2.4 1 SW 4|ADC DB4
P2.3 1 SW 3|ADC DB3
P2.2 1 SW 2|ADC DB2
P2.1 1 SW 1|ADC DB1
P2.0 1 SW 0|ADC DB0
P3.7 1 ADC RD|Comparator Output
P3.6 1 ADC WR
P3.5 1 Motor Sensor
P3.4 1 Display-select Input 1
P3.3 1 AND Gate Output|Display-select 0
P3.2 1 ADC INTR
P3.1 1 Motor Control Bit 1|Ext. UART Rx
P3.0 1 Motor Control Bit 0|Ext. UART Tx

Q.3. A student claims that two assembly programs are equivalent because they access the same RAM address. Using one program with direct addressing and another with indirect addressing, demonstrate why this claim is incorrect by showing a case where both reference the same base address but produce different outputs. Explain how the difference in addressing modes causes this behavior.



Q.4. Write an 8051 Assembly Language Program to generate the last four digits of your mobile number using logical instructions (ANL, ORL, CLR). Do not directly load the complete four-digit number. The final result must be available in the Accumulator (A).



Q.5. Write an 8051 Assembly Language Program to compact data stored in internal RAM locations 40H–5FH using only indirect addressing. Remove all occurrences of FFH, shift the remaining data left to eliminate gaps, and fill the unused locations at the end with 00H, without using additional RAM or the stack.

