

RAMAKRISHNA MISSION VIVEKANANDA
CENTENARY COLLEGE, RAHARA

NAME : DARPAN BHATTACHARYA

COURSE : B.Sc. Computer Science (Hons.)

SEMESTER : 1st Semester

ROLL NUMBER : 715

REGISTRATION NUMBER : A01-1112-117-014-2021 of 2021-2022

SUBJECT : CC2 (Basic Computer Organization and Design)

SESSION : 2021-2022

Problem 1:

Write an Assembly Language Program to add two numbers.

Solution 1:

First operand is stored at location NUM1, second operand is stored at location NUM2, and the sum is stored at SUM.

ALGORITHM:

Step 1: Load augend from NUM1 to Accumulator.

Step 2: Add addend from NUM2.

Step 3: Store the sum present in Accumulator in SUM.

Step 4: Stop the program.

PROGRAM:

LABEL	OPCODE	ADDRESS	COMMENT
	ORG	10	Origin of program is at location 10
NUM1	DEC	5	Decimal augend
NUM2	DEC	6	Decimal addend
SUM	DEC	0	Sum stored in location SUM
	END		End of symbolic program. The sum should be $(11)_{10} = (b)_{16}$.

INPUT DATA	OUTPUT DATA
5, 6	11

SYMBOL	ADDRESS
SUM	11
NUM1	5
NUM2	6

Problem 2:

Write an Assembly Language Program to subtract two numbers.

Solution 2:

Minuend is stored at location NUM1, subtrahend is stored at location NUM2, and the difference is stored at DIFF.

ALGORITHM:

Step 1: Load minuend from NUM1.

Step 2: Add negative subtrahend from NUM2.

Step 3: Store the difference present in Accumulator in DIFF.

Step 4: Stop the program.

PROGRAM:

LABEL	OPCODE	ADDRESS	COMMENT
	ORG	0	Origin of program is at location 0
	CLA		Clear accumulator AC
	LDA	NUM1	Load minuend from NUM1
	ADD	NUM2	Add subtrahend from NUM2
	STA	DIFF	Store difference in location DIFF
	HLT		Halt computer
	ORG	10	Origin of program is 10
NUM1	DEC	54	Decimal minuend
NUM2	DEC	-11	Negative decimal subtrahend
DIFF	DEC	0	Difference stored at location DIFF
	END		End of symbolic program. The difference should be $(43)_{10} = (2b)_{16}$

INPUT DATA	OUTPUT DATA
54, -11	$(43)_{10}$

SYMBOL	ADDRESS
NUM1	54
NUM2	-11
DIFF	0

Problem 3:

Write an Assembly Language Program to multiply two positive numbers through repeated addition.

Solution 3:

Multiplicand is stored at location NUM1, multiplier is stored at location NUM2, and the product is stored at PROD.

ALGORITHM:

Step 1: Load multiplicand NUM1, multiplier NUM2 to memory.

Step 2: Complement accumulator.

Step 3: Increment accumulator to CTR.

Step 4: Store accumulator in CTR.

Step 5: Clear accumulator.

Step 6: Add multiplicand NUM1 to accumulator.

Step 7: Increment CTR by 1.

Step 8: If CTR is not equal to 0, branch to step 6, else continue to next step.

Step 9: Store the product present in accumulator to location PROD.

PROGRAM:

LABEL	OPCODE	ADDRESS	COMMENT
	ORG	0	Origin of program is at location HEX 0
	LDA	NUM2	Load multiplier (6)
	CMA		1's complement of accumulator
	INC		2's complement of accumulator (-6)
	STA	CTR	Store negative of multiplier in CTR
	CLA		Clear accumulator
LOP	ADD	NUM1	Add multiplicand (5, 10, 15, 20, 25, 30)
	ISZ	CTR	Increment counter
	BUN	LOP	Continue loop
	STA	PROD	Store product in location PROD
	HLT		Halt computer

	ORG	30	
NUM1	DEC	5	Multiplier
NUM2	DEC	6	Multiplicand
CTR	DEC	0	Counter (-6, -5, -4, -3, -2, -1, 0)
PROD	DEC	0	Product is stored here
	END		The product should be $(30)_{10} = (001e)_{16}$

INPUT DATA	OUTPUT DATA
5, 6	001e

SYMBOL	ADDRESS
LOP	0
NUM1	5
NUM2	6
CTR	0
PROD	0

Problem 4:

Write an Assembly Language Program to perform subtraction using 2's complement.

Solution 4:

Minuend is stored at location NUM1, subtrahend is stored at location NUM2. Subtract using 2's complement and the difference is stored at DIFF.

ALGORITHM:

Step 1: Load minuend NUM1, subtrahend NUM2 in memory.

Step 2: Load subtrahend NUM2 to accumulator.

Step 3: Complement accumulator.

Step 4: Increment accumulator by 1 for 2's complement of subtrahend.

Step 5: Add minuend NUM1 to accumulator.

Step 6: Store the difference present in accumulator to location DIFF.

PROGRAM:

LABEL	OPCODE	ADDRESS	COMMENT
	ORG	0	Origin of program is at location 0
	CLA		Clear accumulator AC
	LDA	NUM2	Load subtrahend from NUM1
	CMA		Complement AC
	INC		Increment AC
	ADD	NUM1	Add minuend
	STA	DIFF	Store difference in location DIFF
	HLT		Halt computer
	ORG	10	Origin of program is at location 10
NUM1	DEC	100	Decimal minuend
NUM2	DEC	300	Decimal subtrahend
DIFF	DEC	0	Difference stored in location DIFF
	END		End of symbolic program. The difference should be $(-200)_{10} = (ff38)_{16}$

INPUT DATA	OUTPUT DATA
100, 300	$(-200)_{10}$

SYMBOL	ADDRESS
NUM1	100
NUM2	300
DIFF	0

Problem 5:

Write an Assembly Language Program to add an array of 10 numbers using loop.

Solution 5:

Number array is stored starting from location HEX 40, the result of the sum is stored at SUM.

ALGORITHM:

Step 1: Load 10 values in memory starting from location HEX 40.

Step 2: Load starting address of array, that is, HEX 40 in location PTR.

Step 3: Load DEC -10 at location CTR.

Step 4: Add value stored in memory pointed by PTR to accumulator using direct addressing.

Step 5: Increment PTR and CTR by 1.

Step 6: If CTR = 0, branch to step 4 else continue to next step.

Step 7: Store the sum present in accumulator to SUM.

PROGRAM:

LABEL	OPCODE	ADDRESS	ADDRESSING MODE	COMMENT
	ORG	0		Origin of program is at location HEX 0
LOP	ADD	PTR	1	Add operand to AC at location PTR
	ISZ	PTR		Increment pointer
	ISZ	CTR		Increment counter
	BUN	LOP		Repeat loop again
	HLT			Halt computer
PTR	HEX	40		Pointer to array of numbers
CTR	HEX	-10		Counter to add upto 10 numbers
SUM	HEX	0		Sum is stored at location SUM
	ORG	40		Array of 10 numbers starting at location 30
	DEC	1		
	DEC	2		
	DEC	3		

	DEC	4		
	DEC	5		
	DEC	6		
	DEC	7		
	DEC	8		
	DEC	9		
	DEC	-9		
	END			End of symbolic program. The sum should be $(36)_{10}$.

INPUT DATA	OUTPUT DATA
1, 2, 3, 4, 5, 6, 7, 8, 9, -9	$(36)_{10}$

SYMBOL	ADDRESS
LOP	1
PTR	40
CTR	-40
SUM	0