**Basic ALP’s for 8086**

1. **Write an ALP to reverse the string “HELLO”**

Code:

DATA SEGMENT

STRING1 DB "ENTER YOUR STRING HERE ->$"

STRING2 DB "YOUR STRING IS ->$"

STRING3 DB "REVERSE STRING IS ->$"

INSTRING1 DB 20 DUP("$")

RSTR DB 20 DUP("$")

NEWLINE DB 10,13,"$"

N DB ?

S DB ?

DATA ENDS

CODE SEGMENT

ASSUME DS:DATA,CS:CODE

START:

MOV AX,DATA

MOV DS,AX

LEA SI,INSTRING1

;GET STRING

MOV AH,09H

LEA DX,STRING1

INT 21H

MOV AH,0AH

MOV DX,SI

INT 21H

MOV AH,09H

LEA DX,NEWLINE

INT 21H

;PRINT THE STRING

MOV AH,09H

LEA DX,STRING2

INT 21H

MOV AH,09H

LEA DX,INSTRING1+2

INT 21H

MOV AH,09H

LEA DX,NEWLINE

INT 21H

;PRINT THE REVERSE OF THE STRING

MOV AH,09H

LEA DX,STRING3

INT 21H

MOV CL,INSTRING1+1

ADD CL,1

ADD SI,2

L1:

INC SI

CMP BYTE PTR[SI],"$"

JNE L1

DEC SI

LEA DI,RSTR

L2:MOV AL,BYTE PTR[SI]

MOV BYTE PTR[DI],AL

DEC SI

INC DI

LOOP L2

MOV AH,09H

LEA DX,NEWLINE

INT 21H

MOV AH,09H

LEA DX,RSTR

INT 21H

MOV AH,09H

LEA DX,NEWLINE

INT 21H

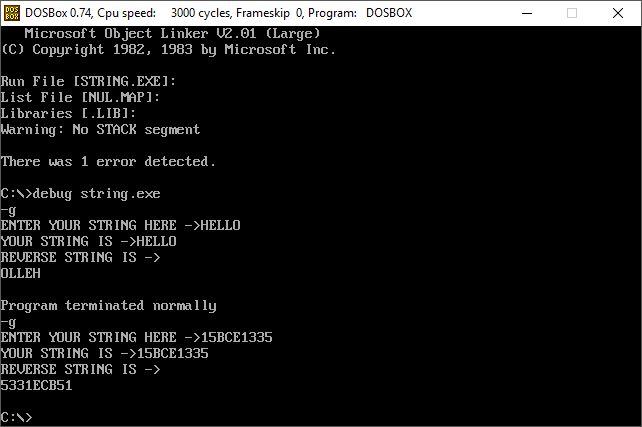
MOV AH,4CH

INT 21H

CODE ENDS

END START

END



1. **Write an ALP to check the given string is palindrome or not (“VITCC”, “MALAYALAM”).**

Code:

DATA SEGMENT

BLOCK1 DB 'MALAYALAM'

MESSAGE1 DB " MALAYALAM: PALINDROME $"

MESSAGE2 DB " MALAYALAM: NOT PALINDROME $"

PAL DB 00H

DATA ENDS

PRINT MACRO MSG

MOV AH,09H

LEA DX,MSG

INT 21H

INT 3H

ENDM

EXTRA SEGMENT

BLOCK2 DB 9 DUP(?)

EXTRA ENDS

CODE SEGMENT

ASSUME CS:CODE,DS:DATA,ES:EXTRA

START: MOV AX,DATA

MOV DS,AX

MOV AX,EXTRA

MOV ES,AX

LEA SI,BLOCK1

LEA DI,BLOCK2+8

MOV CX,00009H

BACK: CLD

LODSB

STD

STOSB

LOOP BACK

LEA SI,BLOCK1

LEA DI,BLOCK2

MOV CX,0009H

CLD

REPZ CMPSB

JNZ SKIP

PRINT MESSAGE1

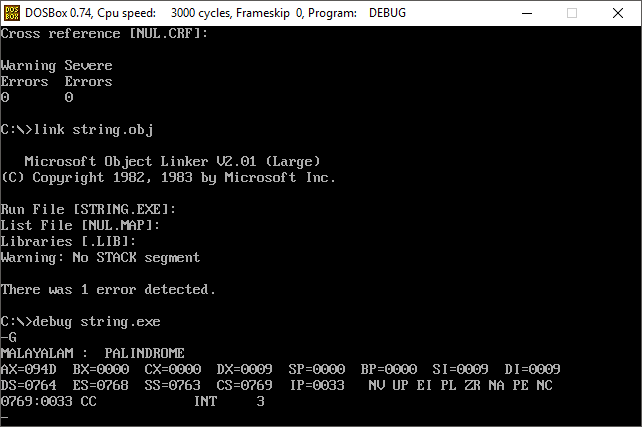
SKIP: PRINT MESSAGE2

CODE ENDS

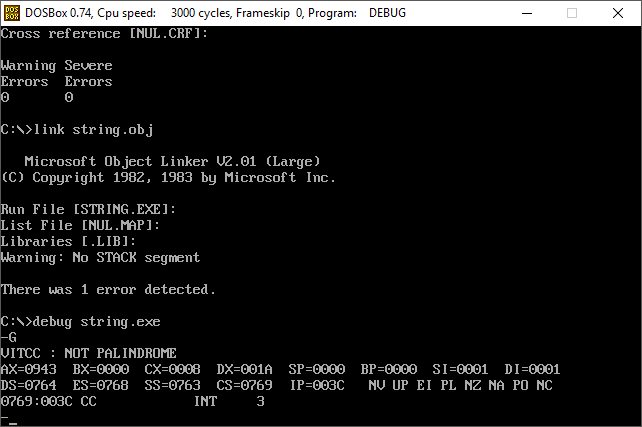
END START

END

CASE: MALAYALAM



CASE: VITCC



1. **Write an ALP to merge two strings: “Hai” and “Hello”**

CODE:

EXTRA SEGMENT

INST1 DB 20 DUP(0)

EXTRA ENDS

DATA SEGMENT

INST2 DB 20 DUP(0)

RESULT DB 40 DUP(0)

MESSAGE1 DB 10,13,'ENTER THE FIRST STRING: $'

MESSAGE2 DB 10,13,'ENTER THE SECOND STRING: $'

MESSAGE3 DB 10,13,'CONCATENATED STRING IS: $'

NEW DB 10,13,'$'

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE,DS:DATA,ES:EXTRA

START:

MOV AX,DATA

MOV DS,AX

MOV AX,EXTRA

MOV ES,AX

LEA DX,MESSAGE1

MOV AH,09H

INT 21H

MOV BX,000

UP1:

MOV AH,01H

INT 21H

CMP AL,0DH

JE DOWN1

MOV [INST1+BX],AL

INC BX

JMP UP1

DOWN1:

LEA DX,NEW

MOV AH,09H

INT 21H

LEA DX,MESSAGE2

MOV AH,09H

INT 21H

MOV CX,BX

MOV BX,00

UP2:

MOV AH,01H

INT 21H

CMP AL,0DH

JE DOWN2

MOV [INST2+BX],AL

INC BX

JMP UP2

DOWN2:

PUSH BX

MOV DI,0

MOV SI,0

UP3:

MOV AL,[INST1+DI]

MOV [RESULT+SI],AL

INC SI

INC DI

LOOP UP3

POP CX

MOV DI,0

UP4:

MOV AL,[INST2+DI]

MOV [RESULT+SI],AL

INC SI

INC DI

LOOP UP4

LEA DX,NEW

MOV AH,09H

INT 21H

LEA DX,MESSAGE3

MOV AH,09H

INT 21H

MOV [RESULT+SI],'$'

LEA DX,RESULT

MOV AH,09H

INT 21H

INT 3

CODE ENDS

END START

END

