	classmate
S.R. POUTA	Date
1BM19CS135	2812 Page
COMPAPE 2 STRING	
	901:100
NODEL SMALL	DEC 84
DISPLAY MACRO MSG	IDA SUZ
LEA DX, MSG	x) 350
mov AH, ORH	PA SITE
THI RIH	143N 1 3099 YHU3Q
ENDM XX 101	XA MUSHO
· DATA XX YOY	OCH CADIAN . tu
MSGI DB ODH, OAH, " ENTER PLI	and the the
MSG 2 DB ODH, OAH, " ENTER SECO	MD STRINGS, 4
MSG 3 DR NOW DAY, " LENGTH DE	2 SIRCING P
MSG 4 DB ODH DAH " LENWIH OF	Security strains
NICK 5 DO DOW ON " STRINGS	ARE EQUAL>
MSG6 DB OOH, OAH, " STRING	IS APT NOT EQUAL)
STRING DB SOH DUP(?)	
STRING 2 DB 80H DUP (?)	
• CoDE	
START: MOV AX, @DATA	
MOU DS, AX	
DISPLAY MSCH	
MOU SE, OFFSET STRING!	
CAU READSTR	
MON BL, CL ; STROP	PE THE LENGITH OF FIRST STRING
DISPLAY MSQ2	
MON SP, OFFSET STRING 2	
CAU READSTR	
ROLPUSH BX	
PUSH CX	
DISPLAY MSQ3	
Mov Al, BL	
CAU LEN_DIS	
DISPLAY MSQ4	
MOV AL, CL	
CAU LEN_PIS	
POP CM POR BX	
POR DA	

CMP CL, BL ; COMPARE THE LENGTHS JNE FAIL ; SE LENGTHS ARE EQUAL, PROCESS MON ST, OFFSET STRING2 CID MON DE, OFFSET STRING2 CHK: MON M, (SS) ; COMPARE BOTH THE STRING2 CMP AL, (DT) JNE FAIL JNC SE HAM MAN MAN MAN MAN MAN MAN MAN MAN MAN M	S.R.BOJA IBMIGESI35	Classed Page	
MOV ST, OFFSET STRINGS MOV DE, OFFSET STRINGS CID CHK: MOV AL (SE) THE FAIL THE ST THE ST THE ST THE ST THE ST THE DEC CL JEVEN MEGS JMP FWAL LEN BS PROC NEAR YOR AH, AH ADD AL, OOH AAM ADD AL, OOH AAM MOV DL, AH MOV DL, BH MOV BH, OOH THE STRINGS JMP FWAL LEN DIS PROC NEAR YOR AH, AL MOV DL, BH MOV BH, OOH THE STRINGS MITTERIAN MOV DL, BH MOV AH, OOH RET LEN DIS ENDP READ STR PRO NEAR YO' YOR CL, CL BACK: MOV AH, OIH TINT 21H	CMP CL BL	, COMPARE THE LENGTHS	
MOV DE, OFFSET STRINGS CHK: MOV AL, (SE) ; COMPARE BOTH THE STEING CMP AL, (DE) JNE PAIL INC ST TINC DE DEC CL JNZ CHK DISPLAY MSGS JMP FWAL LEN_DIS PROC NEAR YOR AH, AH ADD AL, OOH AAM ADD AK, 3030 H MOV BH, AL MOV DL, AH MOV AH, OZH JNT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' YOR CL, CL BACK: MOV AH, OIM INT 21H		, SE LENGTHS ARE EQUAL, P	ROCUSS
CLD CHK: MOU AL, (SE); COMPARE BOTH THE FTENDA CMP AL, (DT) JNE FAIL INC SE INC DE DEC CL JNZ CHK DISPLAY MSGS JMP FUNAL LEN_DIS PROC NEAR YOR AH, AH ADD AL, ODH AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, 02H JOTT 21H RET LEN_DIS ENDP READ STR PRO NEAR YOR AH, OCH AND MOV AH, OCH TNT 21H RET LEN_DIS ENDP READ STR PRO NEAR YOR CL, CL BACK: MON AH, OCH TNT 21H			STATEMENT:
CHK: MOU AL (SE); COMPARE BOTH THE STRING CMP AL (DT) JNE FAIL JNC ST JNC DT DEC CL JNZ CHK DISPLAY MSGS JMP FWAL LEN_DIS PROC NEAR YOR AH, AH ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, 02H JOT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO'R CL, CL BACK: MON AH, OTH TNT 21H	MOV DE OFFSET STRINGS	Tru Beck	
CMP BL, (DT) JNE FAIL JNC ST JNC DI DEC CL JNZ CHK DISPLAY MSGS JMP FWAL LEN_DIS PROC NEAR YOR AH, AH AOD AL, OOH AAM ADD Ax, 3030H MOV BH, AL MOV DL, AH MOV AH, OOH JOT 21H RET LEN_DIS ENDP READ STR PRO NEAR YOR CL, CL BACK: MOV AH, OIM INT 21H	CID , \$1.3118	FINSH: MOV [SE] BYTE	
JNE FAIL JNC ST JNC DE DEC CL JNZ CHK DISPLAY MSGS JMP FWAL LEN_DS PROC NEAR YOR AH, AH ADD AL, DOH AAM ADD Ax, 3030H MOV BH, AL MOV DL, AH MOV AH, D2H JNT 21H RET LEN_DIS ENDP READ STR PRO NEAR YOR CL, CL BACK: MOV AH, O1M INT 21H	CHK: MON AL (38)	, COMPARE BOTH THE STRING	
TINC DE DEC CL JNZ CHK DISPLAY MSGS JMP FIVAL LEN_DIS PROC NEAR YOR AH, AH ADD AL, OOH AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, OZH 3NT 21H RET LEN_DIS ENDP READ STR PRO NEAR YOR CL, CL BACK: MOV AH, OTH TNT 21H		REPOSTR ENDP	
DEC CL DEC CL TINZ CHK DISPLAY MISCAS JMP FLYAL LEN_DIS PROC NEAR YOR AH, AH ADD AH, DOH AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, DOH JOYT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' XOR CL, CL BACK: MOV AH, DIM TINT 21H			
DEC CL JNZ CHK DISPLAY MSG5 JMP FIVAL LEN_DIS PROC NEAR YOR AH, AH ADD AL, ODH AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, OZH JNT 21H PMOV DL, BH MOV AH, DZH JOTT ZHH RET LEN_DIS ENDP READ STR PRO NEAR YO' YOR CL, CL BACK: MON AH, ONH TNT ZHH		FUNK! MOV AH, LICH	-
JNZ CHK DISPLAY MSG5 JMP FLYAL LEN_DS PROC NEAR YOR AHLAH ADD AL, ODH AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AHLOZH 3VT 21H MOV DL, BH MOV AH, D2H INT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' YOR CL, CL BACK: MON AHLONH INT 21H		HUS THE	
DISPLAY MSGS JMP FWAL LEN_DIS PROC NEAR YOR AH, AH ADD AL, ODH AAM ADD AX, 3030 H MOV BH, AL MOV DL, AH MOV AH, OZH JOT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' XOR CL, CL BACK: MOV AH, OTH I'MT 21H		· 79972 QVI	
JMP FINAL LEN_DIS PROC NEAR YOR AH, AH ADD AL, OOH AAM ADD Ax, 3030H MOV BH, AL MOV DL, AH MOV AH, OOH JONT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' XOR CL, CL BACK: MOV AH, OIH INT 21H			
LEN_DIS PROC NEAR YOR AH, AH ADD AL, DOH AAM ADD Ax, 3030H MOV BH, AL MOV DL, AH MOV AH, D2H JOT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' XOR CL, CL BACK: MOV AH, D1H INT 21H			-
YOR AH, AH ADD AH, OOH AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, O2H 3NT 21H RET LEN DIS ENDP READ STR PRO NEAR YS' KOR CL, CL BACK: MOV AH, O1H INT 21H			
ADD AL, ODH AAM ADD AK, 3030 H MOV BH, AL MOV DL, AH MOV AH, ODH JOT 21H RET LEN_DID ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, OTH INT 21H			1
AAM ADD AX, 3030H MOV BH, AL MOV DL, AH MOV AH, 02H JOUT 21H RET LEN DIS ENDP READ STR PRO NEAR YO' XOR CL, CL BACK: MOV AH, 01H ENT 21H			
ADD Ax, 3030H MOV BH, AL MOV DL, AH MOV AH, D2H JOT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, D1H INT 21H			
MOV BH, AL MOV DL, AH MOV AH, O2H MOV AH, O2H INT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, O1H INT 21H			
MOV DL, AH MOV AH, O2H JONT 21H MOV OL, BH MOV AH, O2H JONT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, O1H INT 21H			
MOV AH, 02H BY 21H MOV DL, BH MOV AH, 02H INT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MON AH, O1H INT 21H			-
MOV DL, BH MOV AH, D2H INT 21H RET LEN DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, D1H INT 21H			
MOV DL, BH MOV AH, D2H INT 21H RET LEN_DIS ENDP RETAD STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, OIM INT 21H			
MOV. AH, D2H TOT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MOV AH, D1H TNT 21H			
ENT 21H RET LEN_DIS ENDP READ STR PRO NEAR YO' KOR CL, CL BACK: MON AH, DIM INT 21H			
LEN_DIS ENDP READ STR PRO NEAR YO' YOR CL, CL BACK'. MOU AH, OIH INT 21H			
READSTR PRO NEAR YO' YOR CL, CL BACK: MON AH, DIM INT 21H	RET		
BACK: MON AH, ONH INT 21H	LEN DIS ENDP		
BACK: MON AH, OIH INT 21H	READSTR PRO NEAR		1
INT 21H	19 yor a, a		
CMP AL, DDH			
7 2 0:			
JE BY FINISH	SE SE HNISH		

S.R. POOJA 1BM1905135 MOV (SE), AL MANNO THE LEWISTERS INCOME, PRIOCE LONISTE TERMAL inc a SUNISTE TETA JMP BACK FINISH: MON [SE], BYTE PTR '\$' COMPREE BOTTS THE STEINS. READSTR ENDP PAIL: DISPLAY MSGG FINAL! MON AH, 4CH INT 214 END START.

```
File
                                      Options Help
                                       C:\MASM\STRING.ASM
.MODEL SMALL
DISPLAY MACRO MSG
           LEA DX, MSG
           MOV AH, 09H
           INT 21H
ENDM
.DATA
                                                        : $"
MSG1 DB ODH, OAH, "ENTER FIRST STRING
MSG2 DB ODH, OAH, "ENTER SECOND STRING : $
MSG3 DB ODH, OAH, "LENGTH OF FIRST STRING: $"
MSG4 DB ODH, OAH, "LENGTH OF SECOND STRING: $"
MSG5 DB ODH, OAH, "---STRINGS ARE EQUAL---$"
MSG6 DB ODH, OAH, "---STRINGS ARE NOT EQUAL---$"
STRING1 DB 80H DUP(?)
STRING2 DB 80H DUP(?)
.CODE
START: MOV AX, @DATA
           MOU DS, AX
           DISPLAY MSG1
F1=Help
                                                                       Line:1 Col:1
```

Edit

Search View

File	Edit Search View Opt	
	MOV SI, OFFSET STRING1	:\MASM\STRING.ASM
	CALL READSTR	
	MOV BL, CL	; STORE THE LENGTH OF FIRST STRING
	DISPLAY MSG2	
	MOU SI, OFFSET STRING2	
	CALL READSTR	
	PUSH BX	
	PUSH CX	
	DISPLAY MSG3	
	MOV AL, BL	
	CALL LEN_DIS	
	DISPLAY MSG4	
	MOU AL, CL	
	CALL LEN_DIS	
	POP CX	
	POP BX	
	CMP CL, BL	COMPARE THE LENGTHS
	JNE FAIL	; IF LENGTHS ARE EQUAL, PROCESS NEXT STATMENT
	MOU SI, OFFSET STRING1	
	MOU DI, OFFSET STRING2	
	CLD	
<u>с</u> нк:	MOV AL, [SI]	; COMPARE BOTH THE STRING
F1=Help		Line:44 Col:1

File	Edit Search	View			
e Name (a)	There was Landing		C:\MASM\STRING.ASM		
HK:	MOV AL, [SI]		; COMPARE BOTH THE ST	RING	
	CMP AL, [DI]				
	JNE FAIL				
	INC SI				
	INC DI				
	DEC CL				
	JNZ CHK				
	DISPLAY MSG5				
	JMP FINAL				
EN DIS	PROC NEAR				
	XOR AH, AH				
	ADD AL, OOH				
	AAM				
	ADD AX, 3030H				
	MOV BH, AL				
	MOV DL, AH				
	MOV AH, OZH				
	INT 21H				
	MOV DL, BH				
	MOV AH, 02H				
	INT 21H				
1=Help	THE REST NAME OF TAXABLE PARTY.			Line:65	Col:1

RET LEN_DIS ENDP READSTR PROC NEAR XOR CL, CL BACK: MOV AH, 01H INT 21H CMP AL, ODH JE FINISH MOV [SI], AL INC SI INC CL JMP BACK FINISH: MOV [SI], BYTE PTR '\$' RET READSTR ENDP FAIL: DISPLAY MSG6 FINAL: MOV AH, 4CH INT 21H END START F1-Hel