## TA: Jade Cheng ICS 312 Homework Solution #8

## Debugger: Running the test program in the CodeView debugger

1: Download the file "Test.asm" from the course site, Debugging notes into your masm615/programs directory. I created a folder called hw#8 and put the "Test.asm" in there. Assemble and link the program with command:

C:\WINDOWS\system32\cmd.exe	- 🗆 🗙
C:\masm615\Programs\tests\hw#8>dir Volume in drive C has no label. Volume Serial Number is 6C86-A0A6	
Directory of C:\masm615\Programs\tests\hw#8	
09/15/2009 12:15 AM <dir> 09/15/2009 12:15 AM <dir> 09/14/2009 12:12 PM 749 Test.asm 1 File(s) 7,364,558,048 bytes free</dir></dir>	
C:\masm615\Programs\tests\hw#8≻ml /Zi Test.asm Microsoft (R) Macro Assembler Version 6.14.8444 Copyright (C) Microsoft Corp 1981-1997. All rights reserved.	
Assembling: Test.asm	
Microsoft (R) Segmented Executable Linker Version 5.60.339 Dec 5 1994 Copyright (C) Microsoft Corp 1984-1993. All rights reserved.	
Object Modules [.obj]: Test.obj /CO:nopack Run File [Test.exe]: "Test.exe" List File [rul.map]: NUL Libraries [.16]: Definitions File [rul.def]: CVPACK : warning CK4007: unrecognized option /x; option ignored Microsoft (R) Debugging Information Compactor Version 4.26.01 Copyright (c) Microsoft Corp 1987-1993. All rights reserved.	
C:\masm615\Programs\tests\hw#8>	
	<b>_</b>

As the screen shot shown above, I listed my directory with command "dir" and compiled the "Test.asm" with the flag for debugging information.

Next, run the program in the CodeView Debugger to trace the program. To run the program in CodeView type in the command:

cv Test

File Edit Search Run Data Options Calls Windows Help	ev C:	\WINDOV	VS\system32\cmd.exe - cv Test	- 🗆 🗙
[3]	File	Edit Se	earch Run Data Options Calls Windows Help	
[3]			locals	
[3]       source1_CS:IP_Test.asm         5:       MESSAGE       DB       OOH,OAH, "HULLO WORLD",OOH,OAH, "\$"         6:       X       DW       3         7:       Y       DW       5         8:       Z       DW       7         9:       ARRAY1       DB       10 DUP(2)         10:       .CODE       11:         11:       START       PROC         12:       MOV XX,40ATA       13:         13:       MOV DS,AX       14:         14:       MOV X,42H       :NO USE         15:       MOV X,41H       :NO USE         15:       MOV X,55H       :         16:       MOV Y,CX       :         17:       MOV X,0FPSET MESSAGE         19:       MOV AA!,9         20:       INT 72H         21:       MOV AA',9         20:       INT 72H         21:       MOV AX,4COOH         22:       CALL SU82         23:       MOV AX,4COOH         24:       INT 72H         27:       : A subroutine follows.         (21057 Warning:       CURRENT.STS not found; creating         *				
[3]				
[3]				
[3]				
MESSAGE DB: ODH, OAH, "HULLO WORLD", ODH, OAH, "S" 7: Y DW 5 7: Y DW 7 9: ARRAY1 DB 10 DLP(2) 10: ARRAY1 DB 10 DLP(2) 11: START FROC 12: MOV AX, 40ATA 13: MOV DS, AX 14: MOV DS, AX 15: MOV X, 43H :NO USE 15: MOV X, 43H :NO USE 15: MOV X, 43H :NO USE 16: MOV Y, CX ; 17: MOV Z, 77H ; 18: MOV DY, OFFSET MESSAGE 19: MOV AY, 40OH 20: TMT ZH 21: PLACE: MOV DT,9 ;NO USE 22: CALL SU82 23: MOV AX, 4C0H 24: TMT ZH 25: START ENDP 26: ; 27: ; A subroutine follows. 9: Command	-[3]-		sourcel CS:IP Test asm	
6: X DW 3 7: Y DW 5 8: Z DW 7 9: ARRAY1 DB 10 DUP(2) 10: .CODE 11: START PROC 12: WOV AX, 00ATA 13: WOV OS, AXH; NO USE 13: WOV CX, 55H ; 14: WOV CX, 55H ; 15: WOV CX, 55H ; 16: WOV CX, 55H ; 17: WOV Z, 75H ; 18: WOV Z, 75H ; 19: WOV Z, 75H ; 19: WOV Z, 75H ; 10: WOV Z, 75H ; 11: WOV Z, 75H ; 12: WOV Z, 75H ; 12: WOV Z, 75H ; 13: WOV Z, 75H ; 14: WOV Z, 75H ; 15: WOV Z, 75H ; 15: WOV Z, 75H ; 16: WOV Z, 75H ; 17: WOV Z, 75H ; 17: WOV Z, 75H ; 18: WOV Z, 75H ; 19: WOV Z, 75H ; 10: WOV Z, 75H ; 11: WOV Z, 75H ; 11: WOV Z, 75H ; 12: WOV Z, 75H ; 12: WOV Z, 75H ; 13: WOV Z, 75H ; 14: WOV Z, 75H ; 15: WOV Z, 75H ; 16: WOV Z, 75H ; 17: WOV Z, 75H ; 17: WOV Z, 75H ; 18: WOV Z, 75H ; 19: WOV Z, 75H ; 19: WOV Z, 75H ; 10: W	5:	MESSAGE	DB ODH, OAH, "HULLO WORLD", ODH, OAH, "\$"	
7: Y DW 5 8: Z DW 7 9: ARRAY1 DB 10 DUP(2) 10: .CODE 11: START PROC 12: MOV AX, @OATA 13: MOV DS, AX 14: MOV X, 43H ;NO USE 15: MOV X, 43H ;NO USE 15: MOV X, 45H ; 17: MOV Z, 77H ; 17: MOV Z, 77H ; 17: MOV Z, 77H ; 18: MOV DX, 0FFSET MESSAGE 19: MOV AH, 9 20: INT Z1H 21: PLACE: MOV DI,9 ;NO USE 22: CALL SU82 23: MOV AX, 4C00H 24: INT Z1H 25: START ENDP 26: ; 27: ; A subroutine follows. CUMOSA Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating >■	6:		DW 3	
Si ∠ RAAYI DB 10 DUP(2) 10: .CODE 11: START PROC 12: MOV AX, €0ATA 13: MOV DS, AX 14: MOV CX, 55H ; 15: MOV CX, 55H ; 16: MOV Y, CX ; 17: MOV Z, 77H ; 18: MOV DX, 0FFSET MESSAGE 19: MOV AA, 9 20: INT 21H 20: INT 21H 20: CLU SUB2 20: CLU SU	7:	Y	DW 5	
9: ARCATI DE 10 00+(2) 10: .CODE 11: START PROC 12: MOV AX, @OATA 13: MOV DS, AX 14: MOV X, 41H ;NO USE 15: MOV CX, 55H ; 16: MOV CX, 55H ; 17: MOV Z, 77H ; 17: MOV Z, 77H ; 18: MOV X, 0FSFER MESSAGE 19: MOV AH, 9 20: INT 21H 21: PLACE: MOV DI, 9 ;NO USE 22: CALL SUB2 23: MOV AX, 4COOH 24: INT 21H 25: START ENDP 26: j 27: j A subroutine follows. 29: Command CV1057 Warming: TOOLS.INI not found CV1057 Warming: CURRENT.STS not found; creating >4	8:	Z		
11: START PROC 12: START PROC 12: NOV AX, 40ATA 13: NOV AX, 40ATA 14: NOV X, 43H; NO USE 15: NOV X, 43H; 16: NOV X, 45H; 17: NOV Z, 77H; 17: NOV Z, 77H; 18: NOV AX, 40CH 19: NOV AX, 4COH 21: CALL SUB2 23: NOV AX, 4COH 24: INT 21H 25: START ENDP 26: ; 27: ; A subroutine follows. CVI053 Warning: CURRENT.STS not found; creating >■	10.	ARRATI		
12: WOV AX, QDATA 13: WOV DS, AX 14: WOV X, 41H; NO USE 15: WOV CX, 55H; 16: WOV Y, CX 17: WOV Z, 77H; 17: WOV Z, 77H; 18: WOV DX, OFFSET MESSAGE 19: WOV AH, 9 20: INT Z1H 21: PLACE: WOV DI,9 ;NO USE 22: CALL SUB2 23: WOV AX, 4C00H 24: INT Z1H 25: START ENDP 26: j 27: j A subroutine follows. =0 COMMAND CV1057 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating >	11	START	PROC	
13: MOV DS;AX 14: MOV X,41H ;NO USE 15: MOV CX,55H ; 15: MOV CX,55H ; 17: MOV Z,77H ; 17: MOV Z,77H ; 18: MOV DX,0FFSET MESSAGE 19: MOV AH,9 ;NO USE 20: INT 21H 21: PLACE: MOV DI,9 ;NO USE 22: CALL SU82 23: MOV AX,4C00H 24: INT 21H 25: START ENDP 26: j 27: j A subroutine follows. 19: COMMand CV1057 Warning: COLS.INI not found; creating >>	12:		MOV AX.@DATA	
14:       MOV X,41H ;NO USE         15:       MOV CX,55H ;         16:       MOV Y,CX ;         17:       MOV Z,77H ;         18:       MOV DX,0FFSET MESSAGE         19:       MOV AH.9         20:       INT 21H         21:       PLACE:       MOV DI.9 ;NO USE         22:       MOV AX.4C00H         23:       MOV AX.4C00H         24:       INT 21H         25:       START         27:       ; A subroutine follows.         27:       ; A subroutine follows.         29:       CONMAIN         CV1057 Warning:       COLS.INI not found         CV1057 Warning:       CURRENT.STS not found; creating	13:		MOV DS, AX	
<pre>15: MOV CX,55H ; 16: MOV CY,CX ; 17: MOV Z,77H ; 17: MOV Z,77H ; 17: MOV X,0FFSET MESSAGE 19: MOV AH,9 20: INT 21H 21: PLACE: MOV DI,9 ;NO USE 22: CALL SU82 23: MOV AX,4C00H 24: INT 21H 25: START ENDP 26: ; 27: ; A subroutine follows. 29: Command CV1057 Warning: TOOLS.INI not found; CV1057 Warning: CURRENT.STS not found; creating &gt;&gt;</pre>	14:		MOV X,41H ;NO USE	
Le: MOV T,CX : 1: MOV Z,77H : 13: MOV DX,0FFSET MESSAGE 19: MOV AH.9 20: INT 21H 21: PLACE: MOV DI,9 ;NO USE 22: CAL SUB2 24: MOV AX.4C00H 24: INT 21H 25: START ENDP 26: ; i A subroutine follows. 27: ; A subroutine follows. 29: Command CV1053 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating	15:		MOV CX,55H ;	
A: MOV DX,OFFSET MESSAGE 19: MOV AH,9 20: INT 21H 21: PLACE: MOV DI,9 ;NO USE 22: CALL SUB2 23: MOV AX,4C00H 24: INT 21H 25: 5TART ENDP 26: 27: ; A subroutine follows. START ENDP Command CV1057 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating >	16:			
19; UOV AH;3 DE RESOLUTION 20; INT 21H 21; PLACE: MOV DI.9;NO USE 22; CALL SU82 23; MOV AX,4C00H 24; INT 21H 25; START ENDP 26; ; 27: ; A subroutine follows. 9: Command Construction CV1053 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating	18:		MOV DX.OFESET MESSAGE	
20: INT 21H 21: PLACE: MOV DI.9 ;NO USE 22: CALL SUB 2 23: MOV AX:4C00H 24: INT 21H 25: START ENDP 26: ; A subroutine follows. =[9] Command CV1057 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating >	19:		MOV AH.9	
21: PLACE: MOV DI,9 ;NO USE 22: CALL SUB2 23: MOV AX,4C00H 24: INT 21H 25: j 27: j A subroutine follows. 29: command CV1053 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating	20:		INT 21H	
22: CALL SUB2 23: MOV AX,4COOH 24: INT 21H 25: START ENDP 26: ; 27: ; A subroutine follows. 9: Command CV1053 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating >	21:	PLACE:	MOV DI,9 ;NO USE	
23: MUV AX,4CUUH 24: INT 21H 25: ; ATAT ENDP 26: ; Asubroutine follows. 27: ; A subroutine follows. 29: command CV1053 Warning: TOOLS.INI not found CV1057 Warning: CURRENT.STS not found; creating	22:		CALL SUB 2	
25: START END 26: ; 27: ; A subroutine follows. 9]CVI053 Warning: TOOLS.INI not found CVI057 Warning: CURRENT.STS not found; creating >	23:		TNT 21	
26; ; 27: ; A subroutine follows. (VIO53 Warning: TOOLS.INI not found CVIO57 Warning: CURRENT.STS not found; creating	25:	START	ENDP	
27: ; A subroutine follows. [9]Command CVIO53 Warning: TOOLS.INI not found CVIO57 Warning: CURRENT.STS not found; creating	26:			
=[9]Command CV1053 Warning: TOOL5.INI not found CV1057 Warning: CURRENT.STS not found; creating >■	27:	; A subro	outine follows.	
CVID53 Warning: TOOLS.INI not found CVID57 Warning: CURRENT.STS not found; creating >	101			
CVI057 Warning: CURRENT.STS not found; creating	CV1.05	3 Warning	: TOOLS.INT not found	
	CV105	7 Warning	: CURRENT.STS not found: creating	
-	>			
-				
				-
<trace> <step> <go> <after return=""> <f3=s1 fmt=""> INS DEC</f3=s1></after></go></step></trace>	<trac< td=""><td>e&gt; <step></step></td><td><go> <after return=""> <f3=s1 fmt=""></f3=s1></after></go></td><td>INS DEC</td></trac<>	e> <step></step>	<go> <after return=""> <f3=s1 fmt=""></f3=s1></after></go>	INS DEC

This is the first view I got by executing the command "cv Test". As we saw, we have three windows. They are <u>locals</u>, <u>source1</u>, and <u>command</u>.

In CodeView, arrange the windows as follows:

**a.** Close the <u>command window</u> by clicking on the upper left corner of the window.



**b.** Use the Windows menu to open a <u>Register</u> window and the <u>Memory 1</u> window. The <u>Watch</u> and <u>Source 1</u> windows should already be open. If not, use the Windows menu to add and open them.

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 ×
File         Edit         Search         Run         Data         Options         Calls           [2]         watch         11         11         11         11           [2]         watch         11         11         11         11           [2]         watch         11         11         11         11           [3]	Windows         Halp           Restore         Ctrl+F5           Nove         Ctrl+F7           Tize         Ctrl+F8           Minimize         Ctrl+F8           Minimize         Ctrl+F8           Step         0000           Close         Ctrl+F4           Step         0000           Close         Ctrl+F4           Step         0000           Close         Ctrl+F4           Step         0000           Close         Ctrl+F4           Step         0000           Close         Ctrl+F5           Step         0000           Close         Ctrl+F5           Step         0000           Close         Ctrl+F4           Step         0000           Close         Ctrl+F5           DS         0000           Locals         Alt+F5           Source 1         Alt+3           Sucreat         Alt+6           Submory 1         Alt+5           Submory 1         Alt+5           Subsort         Alt+8           Sucreat         Alt+9           Sucreat         Alt+9     <
L8:         MOV         DX, OFFSET         MESSAGE           19:         MOV         DX, OFFSET         MESSAGE           20:         INT         21H           21:         PLACE:         MOV         DX, OFFSET         MESSAGE           22:         MOV         DX, 0         JY.0         USE           23:         MOV         AX, 4C00H         24:         INT         21H           25:         START         ENDP         26:         .         .         .           26::         :         INT         21H         .         .         .         .           25:         START         ENDP         .         .         .         .         .           26::         :         A subroutime follows.         .         .         .         .           27:         :         A subroutime follows.         .	F0 = f.U≣m==Lie≡ 01 .VWE.Ve§t-cce FF e.ece 00

Use the <u>Windows | Arrange</u> command to get an easy to use layout for the debugger windows.
 (There should be 4 windows open: <u>Watch, Source, Register</u>, and <u>Memory</u>.)

es C:	\WINDO\	WS\syst	em32\cr	nd.exe - c	v Test			- 🗆 🗙
File	Edit S	iearch I	Run Data	0ptions	Calls	Windows	Help	
=[2]=				watch ===			!	[7]reg
								BX = 0000
								CX = 0000
								DX = 0000
							1	BP = 0000
							-i	SI = 0000
[-[3]-	MECCAGE	DP 0	sourcel (	S:IP Test	asm	ALL 11 @ 11		DI = 0000
6:	Х	DW 3	un,uan, n	IULLO WORLL	, oon, o	Ani a		ES = 0A00
7:		DW 5						SS = 0A15
8:	Z	DW ?	0.010(0)					CS = 0A10 TD = 00000
10:	ANNATI	.CODE	0 000 (2)					FL = 0200
11:	START	PROC						NV UP EI PL
12:		MOV	AX, @DATA					NZ NA PO NC
14:		MOV	X.41H :	NO USE				
15:		MOV	CX,55H					
16:		MOV	Y, CX					
18:		MOV	DX.OFFSET	MESSAGE				
19:		MOV	AH,9					
20:	DIACE.	INT	21H	NO USE				
22:	FLACE:	CALL	SUB2	NO USE				
23:		MOV	AX,4COOH					
24:	CTADT	INT	21H					
26:	1 START	ENDF						
27:	A subr	outine	follows.					
-[5]-			- memor	y1 b DS:0	_			
0A00:0	0000 CD	20 FF 9	F 00 9A F	0 FE 1D F	96 02	F0 = f.	Ü≡∎⇔≡ûe≡	
0400:0	0000 07	AB 03 F	0 07 56 0 1 01 FF F	E EE EE EI	4 09 01 E EE EE	EF e.ees	ves4-oee	
0A00:	0027 FF	FF FF F	F FF BB (	9 E8 36 2/	A D2 14	00 _	<b>∃</b> 0Φ6°⊤1.	
0A00:	0034 18	00 00 0	A FF FF F	F FF 00 0	00 00 00	05 10	····+	
040010	0041 00	00 00 0	0 00 00 0	0 00 00 00	00 00			
<pre>Trace</pre>	e> <step></step>	<go> &lt;</go>	After Ret	urn⊳ <f3=< td=""><td>51 Fmt&gt; -</td><td><enter=ex< td=""><td>:pand&gt;</td><td>INS DEC</td></enter=ex<></td></f3=<>	51 Fmt> -	<enter=ex< td=""><td>:pand&gt;</td><td>INS DEC</td></enter=ex<>	:pand>	INS DEC

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 ×
File Edit Search Run Data Options Calls Windows Help	
Add Watch Ctrl+W Delete Watch Ctrl+U Guick Watch Shift+F9 Ereakpoints Ctrl+B	$\begin{array}{l} AX = 0000 \\ BX = 0000 \\ CX = 0000 \\ DX = 0000 \\ SP = 0100 \\ BP = 0000 \end{array}$
	SI = 0000 DT = 0000
S: MESSAGE DB 000H,0AH, "HULLO WORLD", 00H,0AH, "S" 6: X DW 5 8: Z DW 7 8: Z DW 7 10: COE 10: COE 11: START FOO 12: MOV AX @DATA	DS = 0A00 DS = 0A00 SS = 0A10 IP = 0000 FL = 0200 IV UP EI PL Z NA PO NC
11:         MOV         DS, AX           14:         MOV         X, 41H         NO         USE           15:         MOV         X, 41H         ; NO         USE           15:         MOV         X, 55H         ;         i           16:         MOV         Y, 25H         ;         i           17:         MOV         Y, 27H         ;         i           18:         MOV         DX, 0FFSET         MESSAGE         j           19:         MOV         AH, 9         j         i           20:         INT         21H         21:         PLACE:         MOV         DI, 9         ; NO         USE           23:         MOV         AX, 4COH         24:         INT         21H         25:         START         ENDP           26:         :         INT         21H         25:         START         ENDP           27:         :         A subrouting follows.         Subrouting follows.         Subrouting follows.         Subrouting follows.	IZ NA PO NC
0A00:0000     CD     20     FF     9F     00     9A     FO     FE     10     96     02     FO     =     f.U≣∎=±û∈≡       0A00:0000     07     A8     03     FO     FE     10     90     10     +XVE=×VeS+∞ces       0A00:001A     01     00     20     10     FF     FF     FF     FF     e.ese       0A00:0027     FF     FF     FF     FF     FF     e.ese     3006°T1       0A000:0027     FF     FF     FF     FF     FF     e.ese     3006°T1       0A000:0027     FF     FF     FF     FF     FF     00     00     00     00       0A000:0041     180     00     AF     FF     FF     FF     FF     FF       0A00:0041     00     00     00     00     00     00     00     00     00	
<trace> <step> <go> <after return=""> <esc=cancel></esc=cancel></after></go></step></trace>	

Add one at a time by typing in the <u>Add Watch</u> window:

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 ×
File Edit Search Run Data Options Calls Windows Help	- [7]
-[2]	AX = 0000
	BX = 0000 CX = 0000
	DX = 0000 SP = 0100
	BP = 0000
[3] source1 CS: IP Test, asm	DI = 0000
5: MESSAGE DB ODH, OAH, "HULLO WORLD", ODH, OAH, "\$" 6: X DW 3	DS = 0A00 ES = 0A00
7: Y DW 5 8: 7 DW 2	SS = 0A15 CS = 0A10
9: ARRAY1 DB 10 DUP(2)	IP = 0000
11: START PROC	NV UP EI PL
12: MOV AX,@DATA 13: Add Watch	NZ NA PO NC
14: 15: Expression: [X]	
16:	
18: < OK > <cancel> &lt; Help &gt;</cancel>	
20: INT 21H	
21: PLACE: MOV DI,9 ;NO USE 22: CALL SUB2	
23: MOV AX,4C00H 24: TNT 21H	
25: START ENDP	
27: A subroutine follows.	
[5] memory1 b DS:0	
0A00:0000 CD 20 FF 9F 00 9A F0 FE 1D F0 96 02 F0 = j.U≣∎+≞üe≡ 0A00:000D 07 AB 03 F0 07 56 01 15 04 AA 09 01 01 •X♥≣•Ve§+cee	
0A00:001A 01 00 02 01 01 FF FF FF FF FF FF FF €.eee 0A00:0027 FF FF FF FF FF BB 09 E8 36 2A D2 14 00 30¢6°⊤¶.	
0A00:0034 18 00 00 0A FF FF FF FF 00 00 00 00 05 1.0	
<pre><pre><pre><pre><pre><pre><pre><pre></pre></pre></pre></pre></pre></pre></pre></pre>	
· · · · · · · · · · · · · · · · · · ·	

Do it repeatedly to create X, Y, and Z watches. They will show up in the watch panel as the image shown below:

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 ×
File Edit Search Run   Data   Options Calls Windows Help	
■ [2] watch	AX = 0000
Y = 5 Z = 0	BX = 0000 CX = 0000
	DX = 0000
	BP = 0000
	SI = 0000 DI = 0000
5: MESSAGE DB ODH, OAH, "HULLO WORLD", ODH, OAH, "\$"	DS = 0A00 ES = 0A00
7: Y DW 5	SS = 0A15
8: 2 DW ? 9: ARRAY1 DB 10 DUP(2)	CS = 0A10 IP = 0000
10: .CODE	FL = 0200
12: MOV AX, @DATA	NZ NA PO NC
13: MOV DS,AX 14: MOV X,41H ;NO USE	
15: MOV CX,55H 16: MOV Y.CX	
17: MOV Z, 77H	
19: MOV AH,9	
20: INT 21H 21: PLACE: MOV DI.9 :NO USE	
22: CALL SUB 2	
24: INT 21H	
25: START ENDP 26: :	
27: ; A subroutine follows.	
[5] memory1 b DS:0	
0A00:0000 07 AB 03 F0 07 56 01 15 04 AA 09 01 01 •X♥≣•Ve§♦-cee	
0A00:001A 01 00 02 01 01 FF FF FF FF FF FF FF €.eee 0A00:0027 FF FF FF FF FF BB 09 E8 36 2A D2 14 00 ∃c06*⊤1.	
0A00:0034 18 00 00 0A FF FF FF FF 00 00 00 00 05 1.8	
	THE DEC
j <irace> <step> <uo> <atter keturn=""> <f3=51 fmt=""> <enier=expand></enier=expand></f3=51></atter></uo></step></irace>	INS DEC

Use F8 (Trace) to trace the first 2 instructions of the program. This should set a new value in the DS register (in the register window).

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 ×
File Edit Search Run Data Options Calls Windows Help	
2 2 3 WATCH 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	$\begin{array}{c} \text{L7 Jreg} \\ \text{AX} = 0\text{A12} \\ \text{BX} = 0000 \\ \text{CX} = 0000 \\ \text{DX} = 0000 \\ \text{SP} = 0100 \\ \text{BP} = 0000 \\ \text{ST} = 0000 \end{array}$
[-[3]	DI = 0000
5: MESSAGE DB ODH, OAH, "HULLO WORLD", ODH, OAH, "\$"	DS = 0A12
7: Y DW 5	ES = 0A00 SS = 0A15
8: Z DW ?	CS = 0A10
9: ARRAY1 DB 10 DUP(2)	IP = 0005
11: START PROC	NV UP EI PL
12: MOV AX,@DATA	NZ NA PO NO
13: MOV DS, AX	d = 0.01 -
15: MOV X,41H ;NO USE	0003
16: MOV Y,CX	0002
17: MOV Z,77H ;	
19: MOV DA, OFFSET MESSAGE	
20: INT 21H	
21: PLACE: MOV DI,9 ;NO USE	
22: CALL SUB2 23: MOV_AX.4C00H	
24: INT 21H	
25: START ENDP	
27: : A subroutine follows.	
$\frac{1}{10000000000000000000000000000000000$	
0A00:0000 07 AB 03 F0 07 56 01 15 04 AA 09 01 01 •½♥≡•Ve§♦-cee	
0A00:001A 01 00 02 01 01 FF FF FF FF FF FF FF FF @.eee	
0A00:0027 FF FF FF FF BB 09 92 28 63 05 14 00 ¬OÆ(c♠1. 0A00:0034 18 00 00 0A FF FF FF FF 00 00 00 05 t ■	
0A00:0041 00 00 00 00 00 00 00 00 00 00 00 00 00	
<pre></pre> <step> <go> <after return=""> <f3=s1 fmt=""> <enter=expand></enter=expand></f3=s1></after></go></step>	INS DEC

As we saw, DS changed from 0A00 to 0A12. At this point the following instructions in the assembly code were executed:

START

PROC MOV AX,@DATA MOV DS,AX

5:

Click in the upper left corner of the Memory window, and change the first 4 values in the "segment:offset" address information shown to match the value in the DS register.

The first 4 values in the segment address are right now 0A00, and the address in DS is right now 0A12. We would need to change it into the currently used segment to view the contents of memory used by our program.

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 🗙
File Edit Search Run Data Options Calls Windows Help	71.000
X = 3	AX = 0A12
Y = 5 Z = 0	BX = 0000 CX = 0000
	DX = 0000 SP = 0100
	BP = 0000
-[3]	DI = 0000
5: MESSAGE DB ODH, OAH, "HULLO WORLD", ODH, OAH, "\$" 6: X DW 3	DS = 0A12 FS = 0A00
7: Y DW 5	SS = 0A15
9: ARRAY1 DB 10 DUP(2)	IP = 0005
10: .CODE 11: START PROC	FL = 3202 NV UP EI PL
12: MOV AX,@DATA	NZ NA PO NC
14: MOV X,41H ;NO USE	ds:001e
15: MOV CX,55H ; 16: MOV Y,CX ;	0003
17: MOV Z, 77H 18: NOV DX OFFSET MESSAGE	
19: MOV AH, 9	
20: INT 21H 21: PLACE: MOV DI,9 ;NO USE	
22: CALL SUB2 23: MOV AX.4C00H	
24: INT 21H	
26: ;	
27: ; A subroutine follows.	
■=[5] memory1 b 0x0A12:0x0000 [1] 0A12:0000 09 00 E8 05 00 B8 00 4C CD 21 BF 77 00 0.000.3.L=!	
0A12:000D C3 0D 0A 48 55 4C 4C 4F 20 57 4F 52 4C	
0A12:0027 02 02 02 02 02 02 02 02 4E 42 4E 42 30 39 eeeeeeeNBNB09	
0A12:0034 A4 04 00 00 00 00 00 01 01 00 43 56 01 A+	
الم حTrace> <step> <go> <after return=""> <f3=s1 fmt=""> <sh+f3=m1 fmt=""></sh+f3=m1></f3=s1></after></go></step>	INS DEC

Trace a few more instructions using the F8 key and observe the changes to the variables X, Y, and Z in the watch window.

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 ×
File Edit Search Run Data Options Calls Windows Help	-[7]reg
X = 65 Y = 85 Z = 119	AX = 0A12 BX = 0000 CX = 0055
	DX = 00000000000000000000000000000000000
S: MESSAGE DB ODH, OAH, "HULLO WORLD", ODH, OAH, "\$"	DI = 0000 DS = 0A12 ES = 0A00
/: T DW 9 8: Z DW 9 9: ARRAY1 DB 10 DUP(2) 10: .CODE	CS = 0A1S CS = 0A10 IP = 0018 FL = 3202
11: START PROC 12: MOV AX, @DATA 13: MOV DS, AX	NV UP EI PL NZ NA PO NC
14: MOV X,41H ;NO USE 15: MOV CX,55H ; 16: MOV Y,CX ; 17: MOV Z,77H :	
18: MÖV DX,OFFSET MESSAGE 19: MOV AH,9 20: TNT 21H	
21: PLACE: MOV DT.9 ;NO USE 22: CALL SUB2 23: MOV AX,4COOH	
24: INT 21H 25: START ENDP 26: : A submutime fallows	
=[5] remond h 0x0412:0x0000 [1	
0Å12:0000 09 00 E8 05 00 E8 00 4C CD 21 BF 77 00 c.Φ., .L=i,w. 0Å12:000D C3 00 0A 48 55 4C 4C 4F 20 57 4F 52 4C + BHULL 0WRL 0Å12:001A 44 00 0A 24 41 00 55 00 77 00 02 02 02 DDÅ2A.U.w.eee	
0A12:0027 02 02 02 02 02 02 02 4E 42 4E 42 30 39 seceseeNBRBC92 0A12:0034 A4 04 00 00 00 00 00 101 00 43 56 01 m+e.CVs 0A12:0041 00 00 00 00 00 00 02 E 00 00 00 8 54	
   <trace> <step> <go> <after return=""> <f3=s1 fmt=""> <sh+f3=m1 fmt=""></sh+f3=m1></f3=s1></after></go></step></trace>	INS DEC

The values of X, Y and Z were updating as the program executes. At this point the following instructions in the assembly code were executed:

START	PROC		
	MOV	AX,@DATA	<u>x</u>
	MOV	DS,AX	
	MOV	Х,41Н	;NO USE
	MOV	СХ,55Н	;
	MOV	Y,CX	;
	MOV	Z,77H	;
	MOV	DX,OFFSE	T MESSAGE

Once the instruction

7:

MOV DX, OFFSET MESSAGE

has been executed (notice the change to DX in the register window), change the offset of the first address given in the Memory window to match the value stored in DX. You should see the message "Hullo World" in ASCII text at the right end of the Memory Window.

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 🗙
File Edit Search Run Data Options Calls Windows Help 2) X = 65	[7]reg AX = 0A12
Y = 85 Z = 119	BX = 0000 CX = 0055 DX = 0000 SP = 0100
-[3]	BP = 0000 SI = 0000 DI = 0000
5: MESSAGE DB ODH,OAH,"HULLO WORLD",ODH,OAH,"\$" 6: X DW 3 7: Y DW 5	DS = 0A12 ES = 0A00 SS = 0A15
8: Z DW ? 9: ARRAY1 DB 10 DUP(2) 10:CODE	CS = 0A10 IP = 001B FL = 3202
11: START PROC 12: MOV AX,@DATA 13: MOV DS,AX	NV UP EI PL NZ NA PO NC
14: MOV X,41H ;NO USE 15: MOV CX,55H ; 16: MOV Y,CX ; 19: MOV 7,774	
18: MOV DX,OFFSET MESSAGE 19: MOV AH,9 20: INT 21H	
21: PLACE: MOV DT.9 ;NO USE 22: CALL SUB2 23: MOV AX,4C00H	
24: INT 21H 25: START ENDP 26: ;	
27: ; A subroutine follows. =[5]memory1 b 0x0A12:0x000E	
OAL2:0018 00 0A 48 55 4C 4C 4F 20 57 4F 52 4C 4A ♪0+ULLO WORLD OAL2:0018 00 0A 24 41 00 55 00 77 00 02 02 02 02 02 ↓05A.U.w.eeee 0AL2:0228 02 02 02 02 02 02 4E 42 4E 42 30 39 A4 eeeeeeNBNB09ñ	
0A12:0035 04 00 00 00 00 00 01 01 00 43 56 01 00 +ee.CVe. 0A12:0042 00 00 00 00 00 00 02 E0 00 00 08 54 65	
<pre></pre>	INS DEC

The value in offset was 0000. The address contained in DX is 000E. In order to view the contents of the memory space that DX contains, we need to change 0000 to 000E as shown above. After doing that, we can see the message "HULLO WORLD" in ASCII on the right end of the memory window.

Continue tracing the program until the INT 21h instruction has been executed.

C:\WINDOWS\system32\cmd.exe - cv Test	- 🗆 X
File Edit Search Run Data Options Calls Windows Help	
[2]         watch           X = 65         Y           Y = 85         Z           Z = 119	T7 reg AX = 0922 BX = 0000 CX = 0055 DX = 000E SP = 0100 BP = 0000 SI = 0000 DI = 0000 DS = 0413
5: X DW 3 6: X DW 3 7: Y DW 5 8: Z DW 7 9: ARRAY1 DB 10 DUP(2)	ES = 0A12 ES = 0A00 SS = 0A15 CS = 0A10 IP = 001F EV = 3202
101         START         PROC           12:         START         PROCX           12:         MOV         DS, AX           14:         MOV         DS, AX           14:         MOV         X,41H           15:         MOV         CX,55H           16:         MOV         Y,CX           17:         MOV         Z,77H           18:         MOV         DX,0FPST           19:         MOV         AH,9	FL = 3202 NV UP EI PL NZ NA PO NC
21: PLACE: MOV DI.9 ;NO USE 22: CALL SUB2 23: MOV AX,4C00H 24: INT 21H 25: START ENDP	
26: ; 27: ; A subroutine follows.	
Image: State	
<pre>40 41 45 45 45 45 45 45 45 45 45 45 45 45 45</pre>	INS DEC

At this point the following instructions in the assembly code were executed:

START	PROC	
	MOV	AX,@DATA
	MOV	DS,AX
	MOV	Х,41Н
	MOV	СХ,55Н
	MOV	Y,CX
	MOV	z,77н
	MOV	DX,OFFSET MESSAGE
	MOV	АН,9
	INT	21н
PLACE:	MOV	DI,9
	CALL	SUB2
	MOV	АХ,4С00Н
	INT	21н
START	ENDP	
SUB2	PROC	
	MOV	DI,77H ;NO USE
	RET	
SUB2	ENDP	
	END	START

Use the F4 key to view the message displayed on the output screen. Then, use F4 to return to the debugger screen.



We can see that the print-to-screen commend has executed successfully.

9:

Continue tracing the program using F8, including the subroutine, until the program ends.



Click Enter and you are all done!