המרכז לחינוך סייבר
CYBER EDUCATION CENTER

Assembly Workspace Install Guide

Barak Gonen

This tutorial will guide you through installation of the assembly workspace and basic operations.

Tools:

Before we begin it should be pointed that assembly language is closely related to the hardware which is used for executing the commands. Assembly can be written for various CPU manufacturers (Intel, MIPS etc) and CPU register sizes (16 bit, 32 bit, 64 bit). On top of that, even for a given CPU and register size there are different assembly syntaxes, which slightly differ from each other... So we have to focus and make some choices. Our course will use a specific set of assembly, which is commonly used and can be easily adapted to other platforms:

- Intel x86 family CPU
- 32 bit register size
- FASM syntax

As editor, we shall use notepad++. If you prefer other editor that is possible, but notepad++ is very slim in size and loads very fast.

For debugger / disassembler we shall use OllyDbg. Although there are more modern disassemblers, such as IDA, OllyDbg is free and good for starters.

To conclude, we shall need to download and install:

1. Notepad++, for writing our assembly code
2. FASM, for converting the assembly code to machine code (executable file)
3. OllyDbg, for debugging / disassembling

Install Notepad++

Download notepad++ from the following link:

https://notepad-plus-plus.org/downloads/

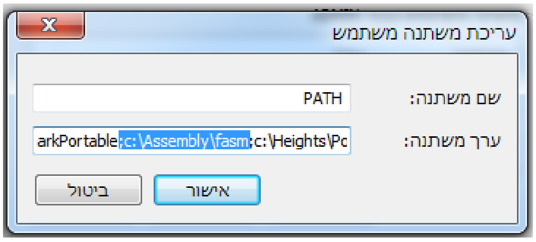
Make sure you select "English" while installing.

Install FASM and OllyDbg

Installations can be downloaded from the following link:

https://data.cyber.org.il/assembly/32bit/installs.zip

- Copy the directory assembly to C:\ (or any other directory, as long as you change the next steps accordingly)
- Edit environment variables (right click My Computer, Properties, Advanced System Settings)
    - To the end of User Variable PATH add ;C:\assembly\fasm (note the semicolon is a must, it separates this entry from the previous one)

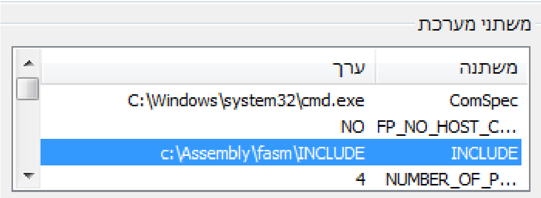


עריכת משתנה משתמש

שם משתנה: PATH

ערך משתנה: arkPortable;c:\Assembly\fasm;c:\Heights\Pc

    - Add a new User Variable called INCLUDE that contains C:\assembly\fasm\include



משתני מערכת

משתנה	ערך
ComSpec	C:\Windows\system32\cmd.exe
FP_NO_HOST_C...	NO
INCLUDE	c:\Assembly\fasm\INCLUDE
NUMBER_OF_P...	4

-

Dedicated Files

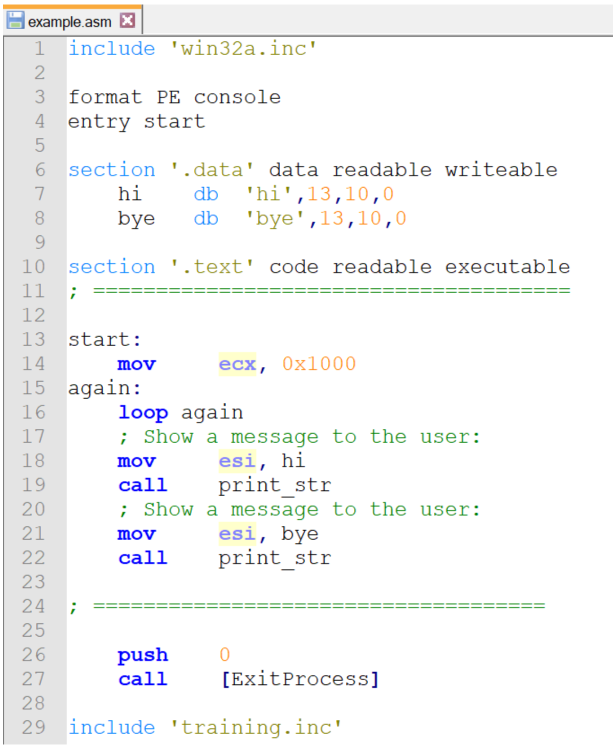
For our course, we shall use two dedicated files which can be found in the “assembly” directory which you have downloaded:

- **example.asm** is an example file which shall be used to demonstrate basic workspace operations.
- **training.inc** is an include file, which adds input/output capabilities to our assembly code.

Using OllyDbg

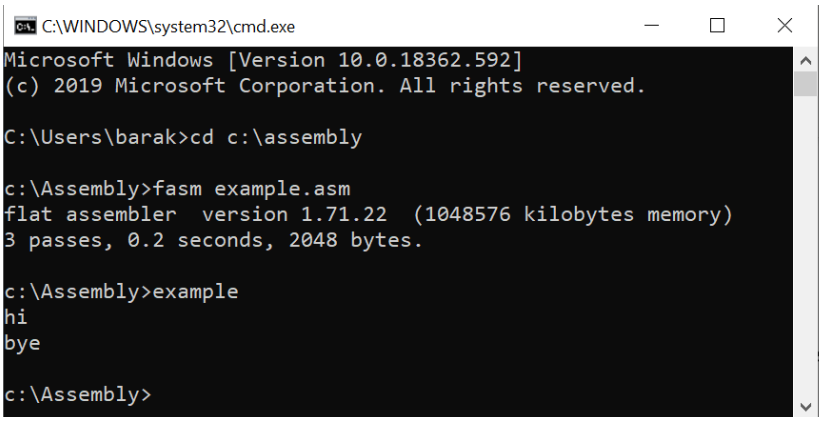
We shall learn to conduct some basic operations

1. Open the file example.asm with notepad++



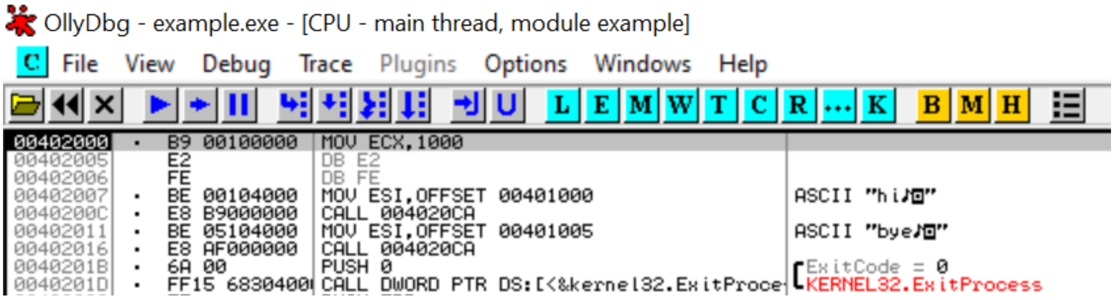
```
include 'win32a.inc'

format PE console
entry start

section '.data' data readable writeable
    hi     db  'hi',13,10,0
    bye    db  'bye',13,10,0

section '.text' code readable executable
; ======================================

start:
    mov     ecx, 0x1000
again:
    loop again
    ; Show a message to the user:
    mov     esi, hi
    call    print_str
    ; Show a message to the user:
    mov     esi, bye
    call    print_str

; ====================================

    push    0
    call    [ExitProcess]

include 'training.inc'
```

2. Use FASM to convert the assembly code into machine language, by writing "fasm example.asm" in the command line. An executable file will be created – example.exe. Run it simply by writing "example", as in the following screenshot:



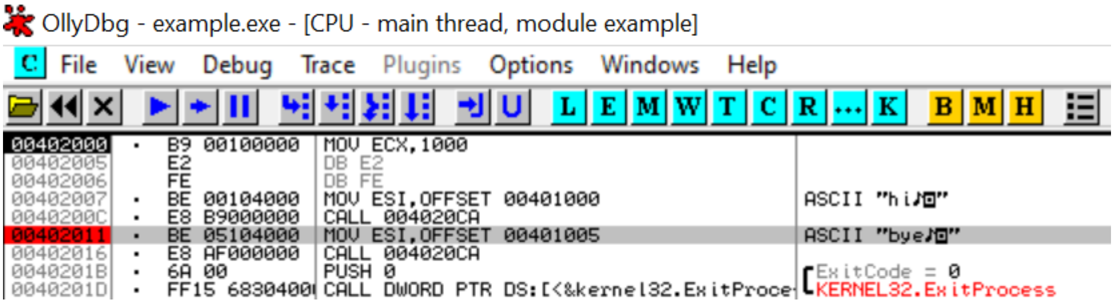
```
Microsoft Windows [Version 10.0.18362.592]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\barak>cd c:\assembly

c:\Assembly>fasm example.asm
flat assembler  version 1.71.22  (1048576 kilobytes memory)
3 passes, 0.2 seconds, 2048 bytes.

c:\Assembly>example
hi
bye

c:\Assembly>
```

3. Use OllyDbg to open the executable file:
   a. In the directory OllyDbg, click on the file "OllyDbg.exe"
   b. Open the file (file->open, then select example.exe from the working directory)
   c. Once opened you will see the following:


OllyDbg - example.exe - [CPU - main thread, module example]
C File View Debug Trace Plugins Options Windows Help
L E M W T C R ... K B M H

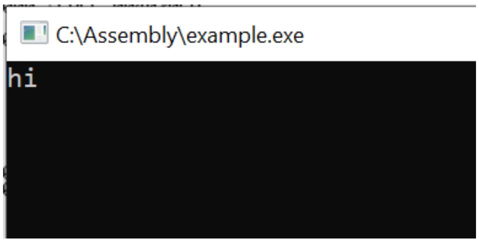
```
00402000  .  B9 00100000  MOV ECX,1000
00402005     E2           DB E2
00402006     FE           DB FE
00402007  .  BE 00104000  MOV ESI,OFFSET 00401000              ASCII "hi♪◘"
0040200C  .  E8 B9000000  CALL 004020CA
00402011  .  BE 05104000  MOV ESI,OFFSET 00401005              ASCII "bye♪◘"
00402016  .  E8 AF000000  CALL 004020CA
0040201B  .  6A 00        PUSH 0                               ┌ExitCode = 0
0040201D  .  FF15 6830400 CALL DWORD PTR DS:[<&kernel32.ExitProce└KERNEL32.ExitProcess
```

   d. Use F7 button to debug step by step
   e. Use F2 to set a breakpoint. Set a breakpoint at the 6th line, as in the screenshot:


OllyDbg - example.exe - [CPU - main thread, module example]
C File View Debug Trace Plugins Options Windows Help
L E M W T C R ... K B M H

```
00402000  .  B9 00100000  MOV ECX,1000
00402005     E2           DB E2
00402006     FE           DB FE
00402007  .  BE 00104000  MOV ESI,OFFSET 00401000              ASCII "hi♪◘"
0040200C  .  E8 B9000000  CALL 004020CA
00402011  .  BE 05104000  MOV ESI,OFFSET 00401005              ASCII "bye♪◘"
00402016  .  E8 AF000000  CALL 004020CA
0040201B  .  6A 00        PUSH 0                               ┌ExitCode = 0
0040201D  .  FF15 6830400 CALL DWORD PTR DS:[<&kernel32.ExitProce└KERNEL32.ExitProcess
```

   f. Use F9 to run until breakpoint is reached. You shall see the following:



```
hi
```

Now that you have completed the installation and basic operation of the workspace, you are good to proceed to the assembly training and tasks ☺ Good luck!