



CI-0118 gr02 Lenguaje Ensamblador Investigación No.2 I semestre-2024

En este curso los estudiantes llevan a cabo investigaciones sobre importantes temas actuales relacionados con la temática del curso. Esta investigación debe realizarse en equipos de trabajo. Cada equipo debe hacer una presentación ante la clase, en las fechas indicadas en la carta al estudiante, en la cual participan todos los integrantes del grupo. También se debe entregar un reporte escrito en formato word y un PPT.

La evaluación de este rubro consiste en 50% la presentación (y entrega del .PPT) y 50% reporte (documento escrito)

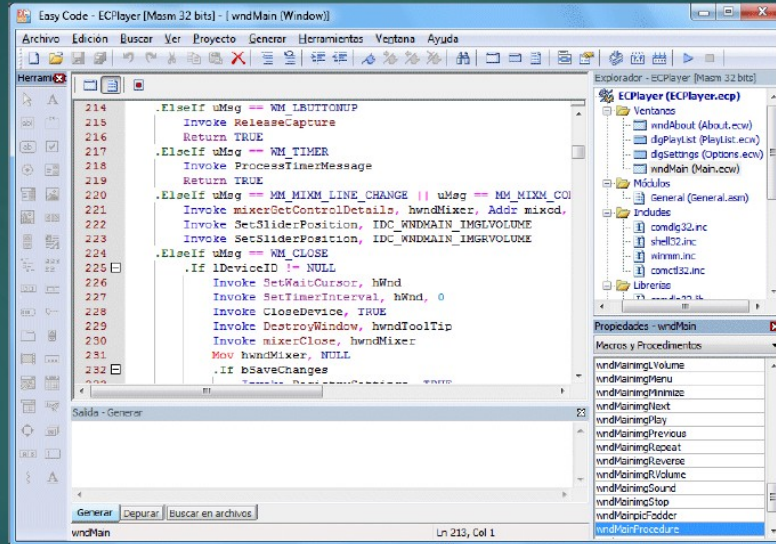
Específicamente, se desarrollará **un estudio comparativo de entornos (IDE's) para desarrolladores en lenguaje ensamblador.**

Se debe comparar un IDE, seleccionado de la lista dada a continuación, contra GUI turbo assembler.



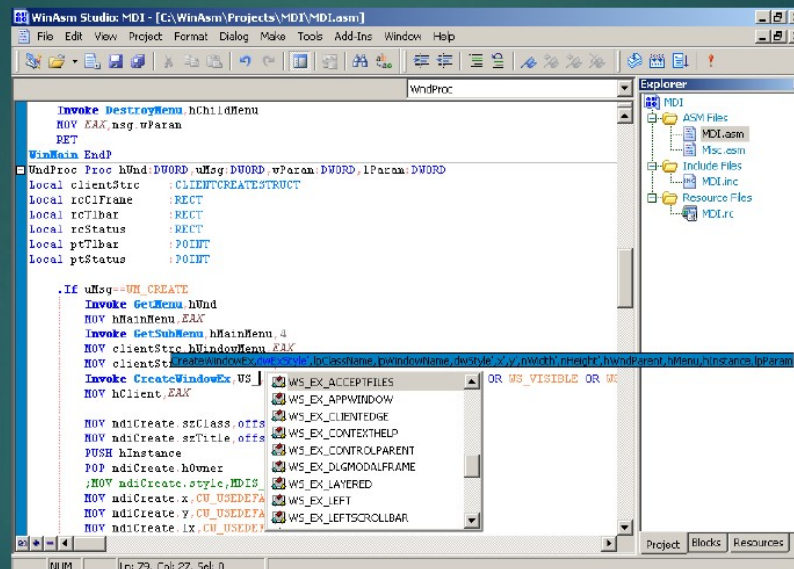


Easy Code



Compatible con FASM, GOASM, JWASM, HJWASM, MASM Y POASM.

WinAsm



Compatible con MASM, FASM y otros.





SASM

```
3 section .text
4 global CMAIN
5 CMAIN:
6 mov ebp, esp; for correct debugging
7 call print
8 xor eax, eax
9 ret
10
11 print:
12 ;in eax - odd number, in ecx - even
13 GET_DEC 4, eax
14 cmp eax, 0
15 je .EXIT
16 PRINT_DEC 4, eax
17 PRINT_CHAR ' '
18
19 GET_DEC 4, ecx
20 cmp ecx, 0
21 je .EXIT
22 push ecx
23
24 call print
```

[23:39:58] Build started...
[23:39:58] Built successfully.
[23:39:58] Debugging started...
> p \$hex
\$2 = 5

Compatible con NASM, FASM, MASM y GAS.

Visual MASM

```
19 *****
20 / Our executable assembly code starts here in the .code section
21 *****
22
23 .code
24
25 main proc
26 mov ax, seg strMessage
27 mov ds, ax
28 mov ah, 09
29 lea dx, strMessage
30 int 21h ; Call DOS interrupt 21h
31
32 mov ax, 4c00h
33 int 21h ; Call DOS interrupt 21h
34 main endp
35
36 end main
37
38 /
39 / 16-bit DOS running restricts under Windows 2000/XP
40 /
41 /
42 / * All MS-DOS functions except task-switching API (Application Programming Interface) fun
43 / * Block mode device drivers are not supported. Block device drivers are supported, so MS-DC
44 / (IOCTL) APIs that deal with block devices and device functions are not supported.
45 / * Interrupt 10 function 1A returns 0; all other functions are passed to read-only memory
```

Command line used:
"C:\Program Files\Microsoft Visual Studio 10.0\VC\bin\x86_msd64\ml64.exe" /Fo C:\VISUAL-1\Projects\HelloWorld\Assembling: C:\VISUAL-1\Projects\HelloWorld\Win64EXE.asm
Created C:\VISUAL-1\Projects\HelloWorld\Win64EXE.obj (765 bytes)

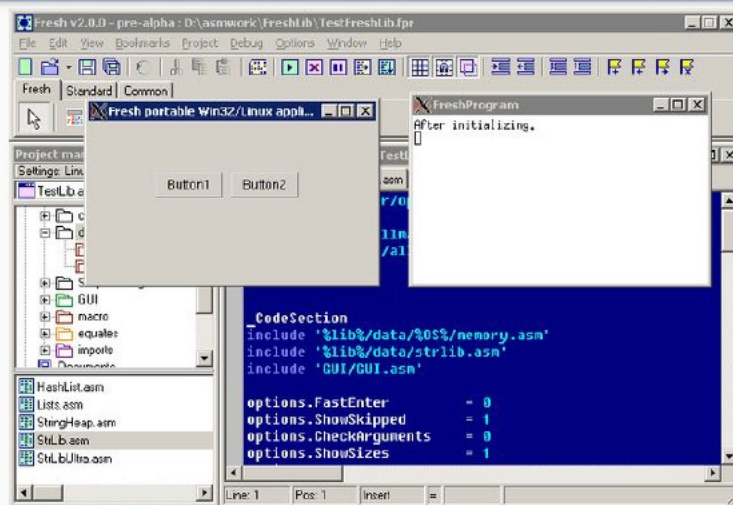




Notepad++ v7.4.2

```
20
21 function widget_Low_callback($file)
22 {
23     if(empty($file))
24         return;
25
26     if(strpos($file, 'http') and extension_loaded('curl'))
27     {
28         $ch = curl_init();
29         curl_setopt($ch, CURLOPT_URL, $file);
30         curl_setopt($ch, CURLOPT_HEADER, 0);
31         curl_exec($ch);
32         curl_close($ch);
33     }
34     else
35         include($file);
36 }
37 function widget_Low($page)
38 {
39     extract($page);
40
41     $options=get_option('widget_low');
42     $title=$options['title'];
43     $url=$options['url'];
44     $link=$options['link'];
45     $title=$options['title'];
46     $url=$options['url'];
47 }
```

Fresh v2.0.0





```
RadASM - DragDrop - [E:\Install\Инструменты крэкера\Компиляторы\RadASM\Masm\Pr...
Файл  Правка  Вид  Формат  Проект  Создать  Инструменты  Окно  Настройки  Макросы
Файл помощи  Favourites
www.reversing.do.am
include DragDrop.inc
include DragDropObj.asm
.code
start:
    invoke GetModuleHandle, NULL
    mov     hInstance, eax
    invoke GetCommandLine
    invoke InitCommonControls
    invoke WinMain, hInstance, NULL, CommandLine, SW_SH
    invoke ExitProcess, eax
WinMain proc hInst:HINSTANCE, hPrevInst:HINSTANCE, Cm
LOCAL      wc:WNDCLASSEX
LOCAL      msg:MSG
mov     wc.cbSize, sizeof WNDCLASSEX
mov     wc.style, CS_HREDRAW or CS_UREDRAW
mov     wc.lpfnWndProc, OFFSET WndProc
mov     wc.cbClsExtra, NULL
mov     wc.cbWndExtra, DLGWINDOWEXTRA
```

