

Práctica 5

Generación de un Programa Ejecutable

Luis Eduardo Galindo Amaya (1274895)

Asignatura	Organización de Computadoras (331)
Docente	Arturo Arreola Alvarez
Fecha	21-09-2022

Generación de un Programa Ejecutable

Luis Eduardo Galindo Amaya (1274895)

21-09-2022

Código

codigo_base.asm

```
1 |
2 | %include "./pc_io.inc"
3 |
4 | section .text
5 | global _start
6 |
7 | _start:                                ; TELL LINKER ENTRY POINT
8 |     mov edx, msg
9 |     call puts
10 |
11 |     mov eax, 1                         ; SYSTEM CALL NUMBER (SYS_EXIT)
12 |     int 0x80                          ; CALL KERNEL
13 |
14 | section .data
15 |     ;; OUR DEAR STRING
16 |     msg db 'Hello , world!', 0xA, 0x0
```

Compilando el Ejemplo

```
nasm -f elf codigo_base.asm
ld -m elf_i386 codigo_base.o libpc_io.a -o salida.out
./salida.out
```

Hello, world!

Imprimir tres strings

Modificar código `_base.asm` y nombrarlo `Apellido_Nombre_P5.asm`. Modificar el código para que imprima al menos 3 cadenas diferentes. Las cadenas debes ser definidas de la misma forma como se hace en `código_base.asm`.

```
1 |
2 | %include "./pc_io.inc"
3 |
4 | section .text
5 | global _start
6 |
7 | _start:
8 |     mov edx,msg0           ; HELLO WORLD
9 |     call puts
10 |
11 |     mov edx,msg1          ; OTRO STRING
12 |     call puts
13 |
14 |     mov edx,msg2          ; Y OTRO
15 |     call puts
16 |
17 |     mov eax,1             ; TERMINAR PROGRAMA
18 |     int 0x80
19 |
20 | section .data
21 |     msg0 db 'Hello , world!', 0xA, 0x0
22 |     msg1 db 'Otro String!', 0xA, 0x0
23 |     msg2 db 'y otro String!', 0xA, 0x0
```

Compilación

```
nasm -f elf Galindo_Amaya_Luis_Eduardo.P5.asm
ld -m elf_i386 Galindo_Amaya_Luis_Eduardo.P5.o libpc_io.a -o salida2.out
./salida2.out
```

```
Hello, world!
Otro String!
y otro String!
```

Captura con getch y imprime con putchar

Dentro del mismo código, añade la opción de capturar un carácter (vea getch) y justo después, imprimirlo (vea putchar).

```
1
2 %include "./pc_io.inc"
3
4 section .text
5     global _start
6
7 _start:
8     mov edx, msg0
9     call puts
10
11     call getch                ; CAPTURA CARACTER
12     push eax
13     call putchar              ; MUESTRA CARACTER
14
15     mov edx, msg1
16     call puts
17
18     mov eax, 1                ; TERMINAR PROGRAMA
19     int 0x80
20
21 section .data
22     msg0 db 'Captura caracter', 0xA, 0x0
23
24     ;; TERMINADOR DE CADENA
25     msg1 db 0xA, 0x0
```

Conclusiones y dificultades

Nasm es mas difícil que marie.js, tiene operaciones mas avanzadas pero requiere que pongamos mas de nuestra parte para entender que estamos haciendo.

Capturas

The screenshot shows an Emacs editor window titled 'emacs@galindo-HP-Notebook'. The left pane displays a LaTeX document with the following content:

```
<tos/ODC-2022/práctica-5
+ ..|
+ src/
Galindo_Amaya_Luis_Eduardc
Galindo_Amaya_Luis_Eduardc
Práctica 5.org
Práctica 5.pdf
Práctica 5.tex
captura-p5.asm
codigo_base.asm
codigo_base.pdf
codigo_base.tex
libpc_io.a
pc_io.inc
salida.out
salida2.out
salida3.out

#*TITLE: Generación de un Programa Ejecutable
#*AUTHOR: Luis Eduardo Galindo Amaya (1274895)
#*DATE: 21-09-2022

#*MACRO: tagline Práctica 5
#+include: "../includes/template.org"
#+options: ^:nil

+ Código ...
+ Imprimir tres strings...
+ Captura con getch y imprime con putchar
Dentro del mismo código, añade la opción de capturar un carácter (v
ea getch) y justo después, imprimirlo (vea putchar).

{{{code(./captura-p5.asm)}}}

\pagebreak
```

The right pane shows the terminal output of the compilation process:

```
galindo@galindo-HP-Notebook:práctica-5$ ./salida3.out
Captura caracter
a
a
galindo@galindo-HP-Notebook:práctica-5$ galindo@galindo-HP-Notebook:
práctica-5$ ./salida3.out
./salida3.out
Captura caracter
e
e
galindo@galindo-HP-Notebook:práctica-5$ galindo@galindo-HP-Notebook:
práctica-5$ ./salida2.out
./salida2.out
Hello, world!
Otro String!
y otro String!
galindo@galindo-HP-Notebook:práctica-5$
```

The status bar at the bottom indicates the file is 'Práctica 5.org', line 56, column 0. The system clock shows 07:35 PM.

The screenshot shows an Emacs editor window titled 'emacs@galindo-HP-Notebook'. The left pane displays a LaTeX document with the following content:

```
<tos/ODC-2022/práctica-5
+ ..
+ src/
Galindo_Amaya_Luis_Eduardc
Galindo_Amaya_Luis_Eduardc
Galindo_Amaya_Luis_Eduardc
Práctica 5.org
Práctica 5.pdf
Práctica 5.tex
captura-p5.asm
codigo_base.asm
codigo_base.pdf
codigo_base.tex
libpc_io.a
pc_io.inc
salida.out
salida2.out
salida3.out

#*TITLE: Generación de un Programa Ejecutable
#*AUTHOR: Luis Eduardo Galindo Amaya (1274895)
#*DATE: 21-09-2022

#*MACRO: tagline Práctica 5
#+include: "../includes/template.org"
#+options: ^:nil

+ Código ...
+ Imprimir tres strings...
+ Captura con getch y imprime con putchar
Dentro del mismo código, añade la opción de capturar un carácter (v
ea getch) y justo después, imprimirlo (vea putchar).

{{{code(./captura-p5.asm)}}}

\pagebreak
```

The right pane shows the terminal output of the compilation process:

```
galindo@galindo-HP-Notebook:práctica-5$ ./salida3.out
Captura caracter
a
a
galindo@galindo-HP-Notebook:práctica-5$ galindo@galindo-HP-Notebook:
práctica-5$ ./salida3.out
./salida3.out
Captura caracter
e
e
galindo@galindo-HP-Notebook:práctica-5$ galindo@galindo-HP-Notebook:
práctica-5$
```

The status bar at the bottom indicates the file is 'Práctica 5.org', line 56, column 0. The system clock shows 07:31 PM.