

Mòdul 05. Entorns de desenvolupament

1r CFGS Desenvolupament d'Aplicacions Multiplataforma / Web

Curs: 2022/2023

UF1. Pt1a – Traducció de llenguatges (part A)

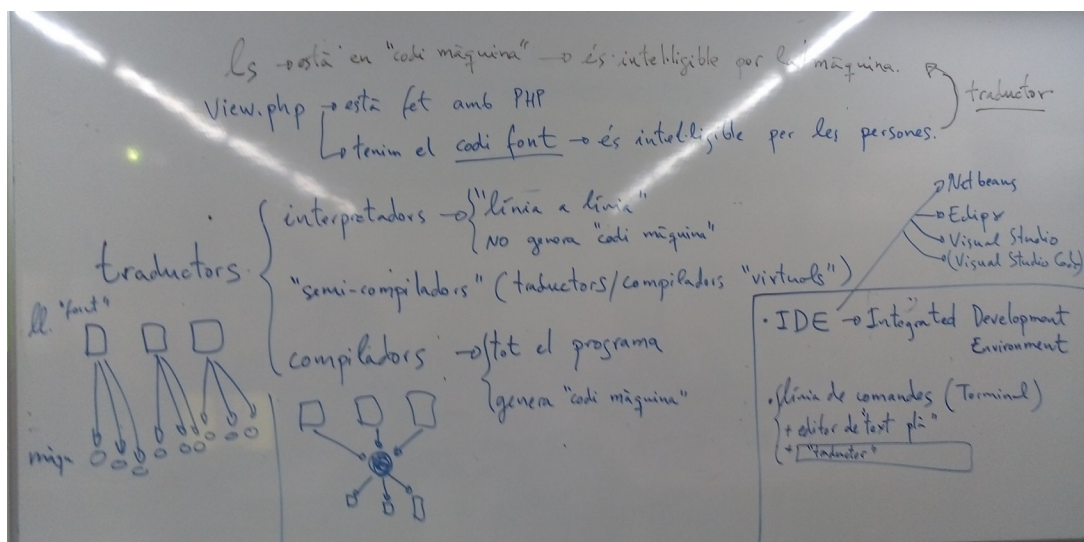


Figura 1: Traductors de llenguatges (entorns de desenvolupament)

El **codi font** és intel·ligible per les persones (i no per l'ordinador).

El **codi màquina** és intel·ligible per l'ordinador (i no per les persones).



Figura 2: Margaret Hamilton

```

174 VRTSTART TS WCHVERT
175 # Page 801
176 CAF TWO # WCHPHASE = 2 ---> VERTICAL: P65,P66,P67
177 TS WCHPHOLD
178 TS WCHPHASE
179 TC BANKCALL # TEMPORARY, I HOPE HOPE HOPE
180 CADR STOPRATE # TEMPORARY, I HOPE HOPE HOPE
181 TC DOWNFLAG # PERMIT X-AXIS OVERRIDE
182 ADRES XOVINFLG
183 TC DOWNFLAG
184 ADRES REDFLAG
185 TCF VERTGUID
186

```

Figura 3: Una pàgina de codi (assemblador). Projecte Apollo 11 Webgrafia:

<https://qz.com/726338/the-code-that-took-america-to-the-moon-was-just-published-to-github-and-its-like-a-1960s-time-capsule/>

<https://github.com/chrislgarry/Apollo-11/blob/master/README.ca.md>

https://ca.wikipedia.org/wiki/Margaret_Hamilton

Què volem fer?

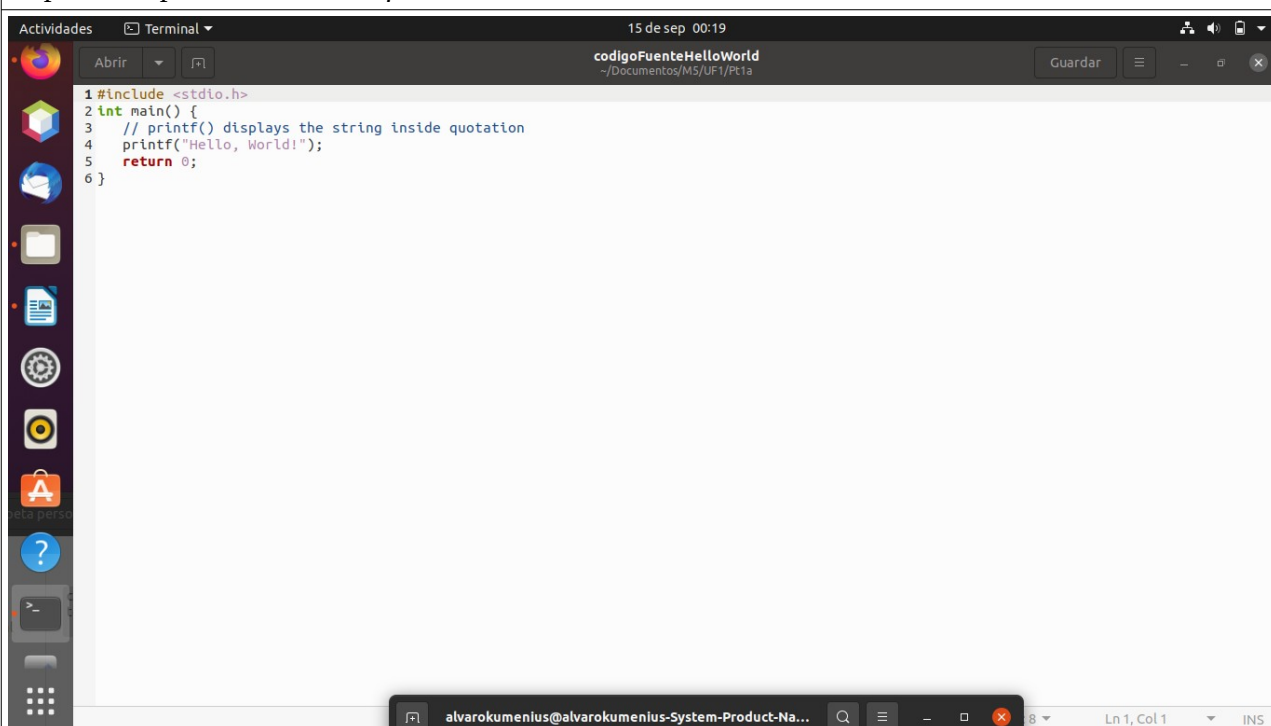
1. Compilar un programa (el codi font està en C) i generar l'executable (codi màquina).
2. Executar el codi màquina generat i veure el resultat.
3. Examinar el programa executable per veure que és intel·ligible per les persones.

Què necessitem?

1. Un **editor** de text pla (per exemple `gedit` o `Kate`).
2. El **codi font** del programa (el podem copiar **d'aquí**¹).
3. El compilador **gcc** (vegeu <https://itsfoss.com/run-c-program-linux/>).
4. El programa **gdb** (vegeu **aquí**² per instal·lar-lo i **aquí**³ per exemples).
5. El programa **okteta** (vegeu **aquí**⁴ per instal·lar-lo i **aquí**⁵ per un exemple).

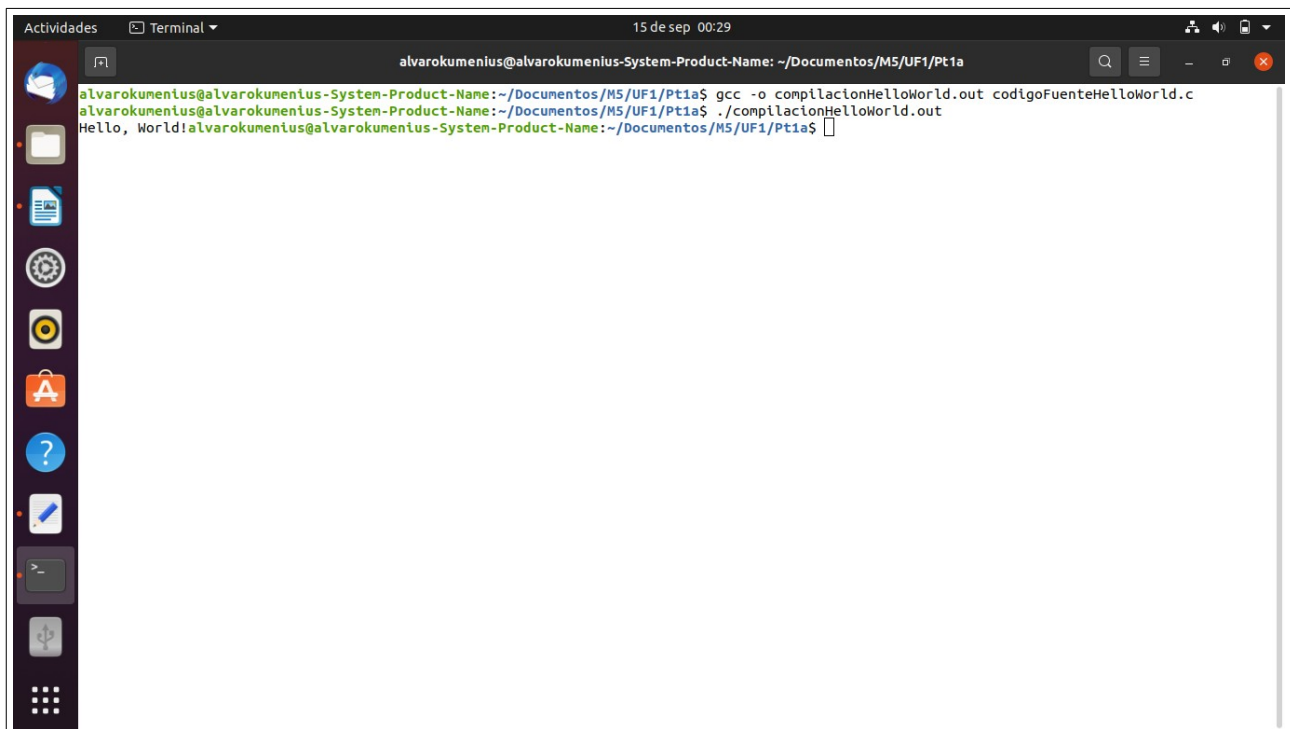
Què hem de lliurar?

Captura de pantalla del codi font en el vostre editor



Captura de pantalla del terminal (fons blanc, si us plau!) amb la compilació i l'execució

- 1 <https://www.programiz.com/c-programming/examples/print-sentence>
- 2 https://lihuen.linti.unlp.edu.ar/index.php/Cómo_usar_GDB
- 3 <https://visualgdb.com/gdbreference/commands/disassemble>
- 4 <https://www.shorttutorials.com/apt-get-commands/install-okteta.html>
- 5 <https://medium.com/@Andromeda/fun-with-binaries-6a057b4ab5ca> (primer exemple només).

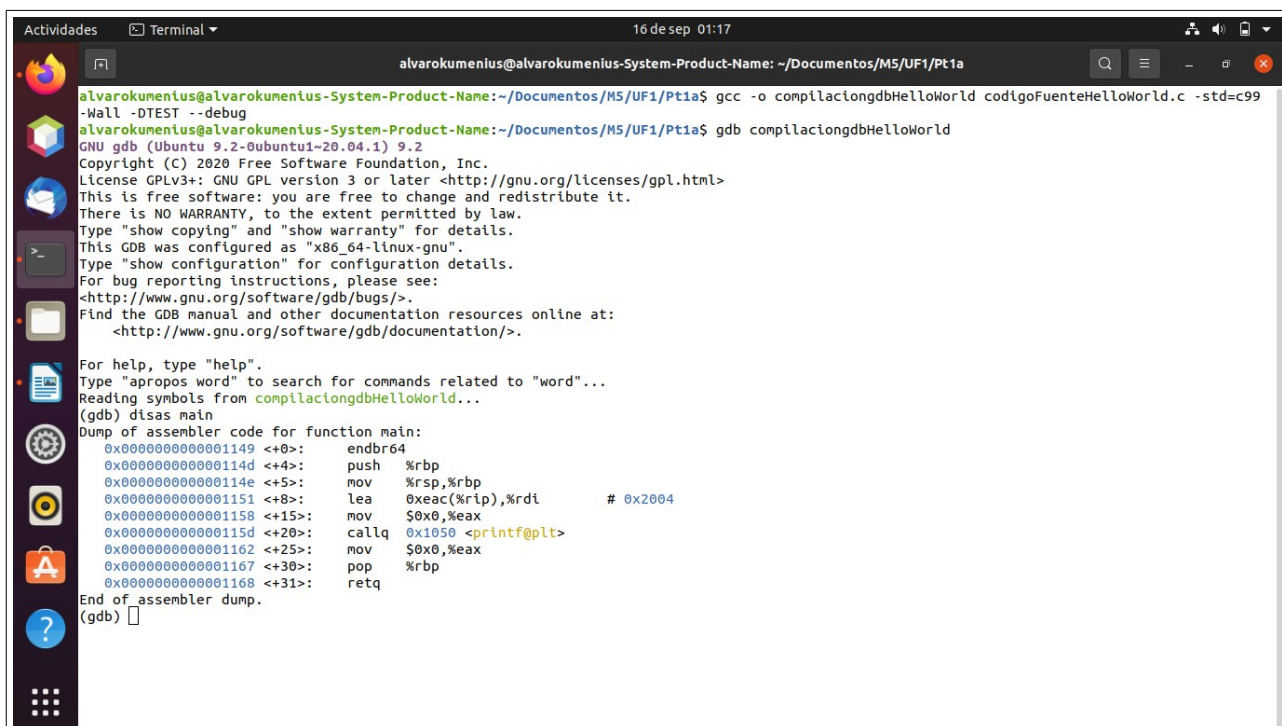


The screenshot shows a terminal window titled 'Terminal' with the date '15 de sep 00:29'. The user is 'alvarokumenius' and the system is 'alvarokumenius-System-Product-Name'. The current directory is '~/Documentos/M5/UF1/Pt1a'. The terminal shows the following commands and output:

```
alvarokumenius@alvarokumenius-System-Product-Name: ~/Documentos/M5/UF1/Pt1a$ gcc -o compilacionHelloWorld.out codigoFuenteHelloWorld.c
alvarokumenius@alvarokumenius-System-Product-Name: ~/Documentos/M5/UF1/Pt1a$ ./compilacionHelloWorld.out
Hello, World!alvarokumenius@alvarokumenius-System-Product-Name: ~/Documentos/M5/UF1/Pt1a$
```

- 1) Guardar el archivo del programa con la extensión «.c».
- 2) Acceder al terminal.
- 3) Introducir comando «gcc -o <nombre del archivo compilado que nos creará>.out <nombre del archivo del programa con la extensión .c que queremos que sea compilado>.c».
- 4) Como resultado, aparece en la misma carpeta el archivo con el código fuente compilado.
- 5) Introducir comando «./<nombre del archivo compilado>.out».
- 6) Como resultado, aparece en el terminal el programa ejecutado.

*Captura de pantalla del codi màquina i assembler (obtingut amb **gdb**)*

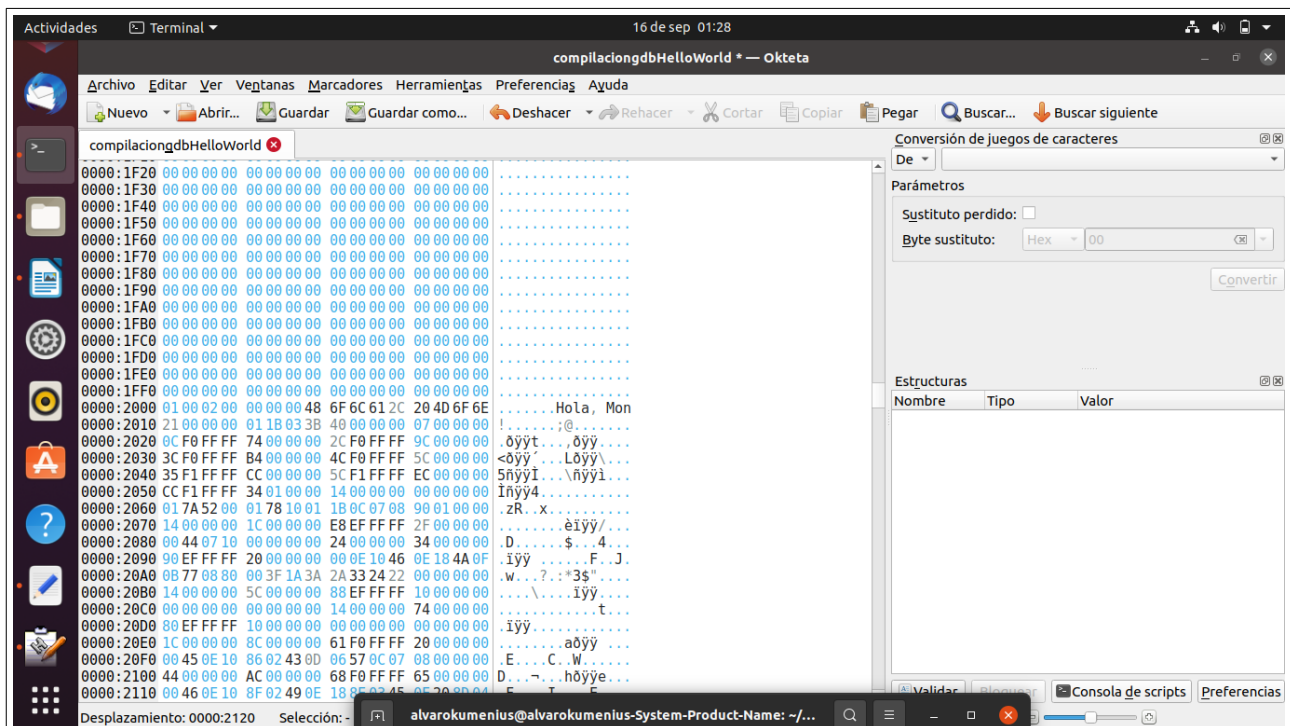


```
alvarokumenius@alvarokumenius-System-Product-Name: ~/Documentos/M5/UF1/Pt1a$ gcc -o compilaciongdbHelloWorld codigoFuenteHelloWorld.c -std=c99 -Wall -DTEST -debug
alvarokumenius@alvarokumenius-System-Product-Name: ~/Documentos/M5/UF1/Pt1a$ gdb compilaciongdbHelloWorld
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from compilaciongdbHelloWorld...
(gdb) disas main
Dump of assembler code for function main:
0x0000000000001149 <+0>:    endbr64
0x000000000000114d <+4>:    push    %rbp
0x000000000000114e <+5>:    mov     %rsp,%rbp
0x0000000000001151 <+8>:    lea     0x00000000(%rip),%rdi    # 0x2004
0x0000000000001158 <+15>:   mov     $0x0,%eax
0x000000000000115d <+20>:   callq   0x1050<printf@plt>
0x0000000000001162 <+25>:   mov     $0x0,%eax
0x0000000000001167 <+30>:   pop     %rbp
0x0000000000001168 <+31>:   retq
End of assembler dump.
(gdb) 
```

- 1) Acceder al terminal.
- 2) Introducir comando «gcc -o <nombre del archivo compilado que nos creará> <nombre del archivo del programa con la extensión .c que queremos que sea compilado>.c -std=c99 -Wall -DTEST -debug». **No cal «-std=c99 -Wall -DTEST -debug»**
- 3) Como resultado, aparece en la misma carpeta el archivo con el código fuente compilado.
- 4) Introducir comando «gdb <nombredelarchivocompilado>»
- 5) En el terminal, después de un bloque de mensaje, aparece «(gdb)». Introducir el comando «disas main» para visualizar el ~~código máquina~~ y el ensamblador.

*Captura de pantalla del codi hexadecimal modificat (amb **okteta**)*



- 1) Archivo > Abrir > Seleccionar el archivo con el programa compilado.
- 2) Aparece el código hexadecimal.
- 3) Ctrl + F para buscar los caracteres «Hello, World».
- 4) Cambiar los caracteres.
- 5) Fer clic a «Guardar»
- 6) Comprovar que el fitxer executable escriu el missatge canviat (vegeu a continuació)

