

Mòdul 05. Entorns de desenvolupament

1r CFGS Desenvolupament d'Aplicacions Multiplataforma / Web

Curs: 2023/2024

UF1. Pt1a – Traducció de llenguatges (part A)

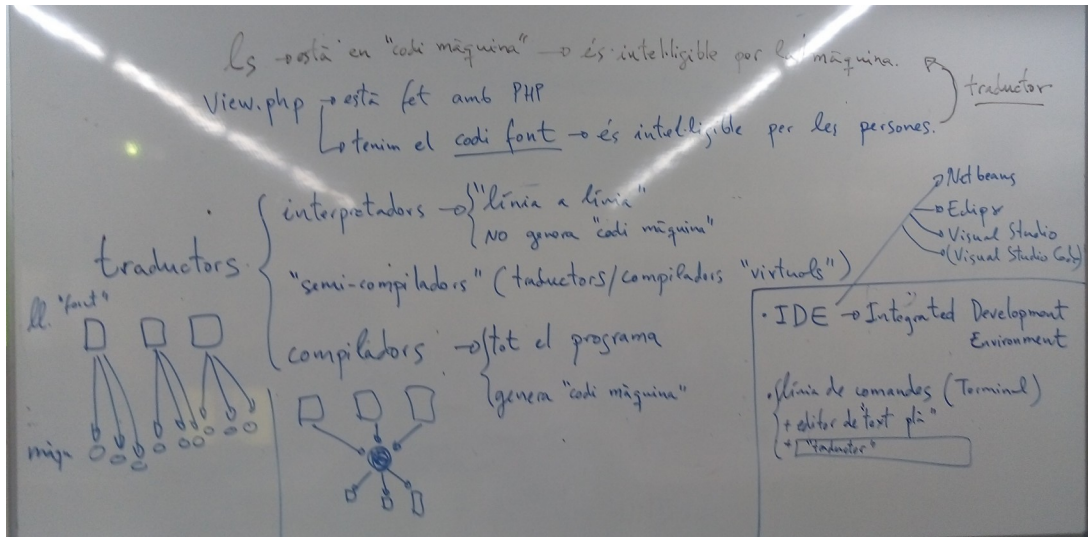


Figura 1: Traductors de llenguatges (entorns de desenvolupament)

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 (fons blanc, si us plau!) del codi font en el vostre editor

Aquest codi font està creat amb un conjunt d'instruccions en un llenguatge creat per les persones i que per tant la màquina no ho entendreà. Aquest codi font (intel·ligible només pels humans) haurà de ser compilat per a que la màquina ho llegeixi i sigui executat.

```
1 #include <stdio.h>
2 int main() {
3     // printf() displays the string inside quotation
4     printf("Hello, World!");
5     return 0;
6 }
```

1 <https://www.programiz.com/c-programming/examples/print-sentence>

2 https://lihuen.linti.unlp.edu.ar/index.php/Cómo_usar_GDB

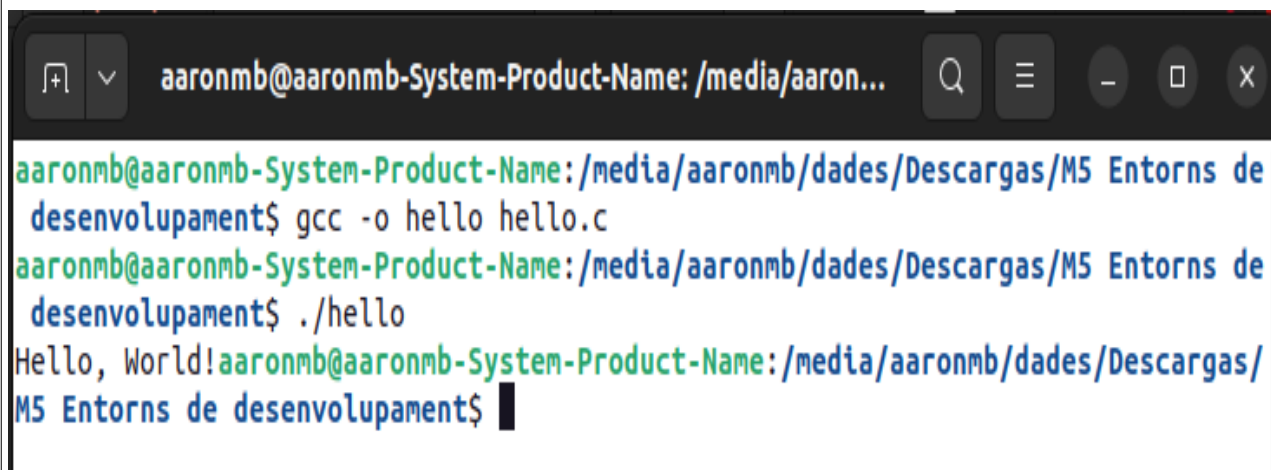
3 <https://visualgdb.com/gdbreference/commands/disassemble>

4

5

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

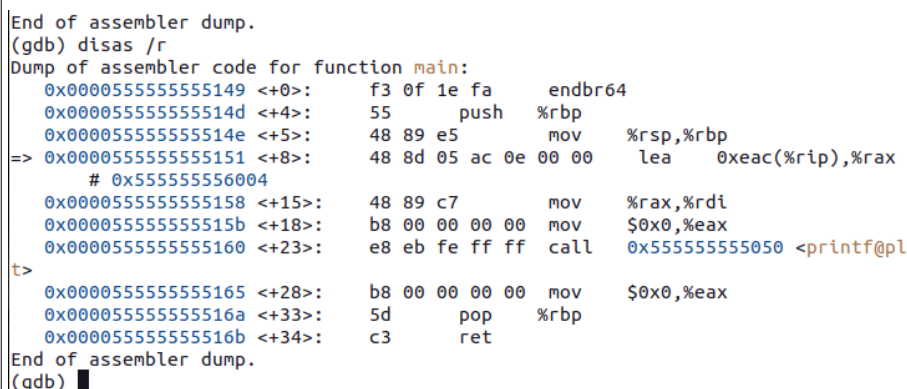
Per tant el que hem fet aquí per executar el nostre programa informàtic es utilitzar en la terminal un compilador (gcc) per passar del llenguatge C (codi font) al llenguatge executable (codi màquina), i així fer que el procesador executi el codi compilat instrucció per instrucció. En aquest cas veiem el Hello, World! (que es el que volíem fer).



```
aaronmb@aaronmb-System-Product-Name: /media/aaron...
aaronmb@aaronmb-System-Product-Name: /media/aaronmb/dades/Descargas/M5 Entorns de desenvolupament$ gcc -o hello hello.c
aaronmb@aaronmb-System-Product-Name: /media/aaronmb/dades/Descargas/M5 Entorns de desenvolupament$ ./hello
Hello, World!aaronmb@aaronmb-System-Product-Name: /media/aaronmb/dades/Descargas/M5 Entorns de desenvolupament$
```

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

Ara, amb el gdb, que és un depurador que ens permet inspeccionar el codi realitzat i que ens pot detectar el errors més ràpid del programa, ens mostra el codi assemblador i el codi màquina del nostre programa informàtic. Ho hem fet amb el comand disassemble de gdb a la terminal (disas /r)

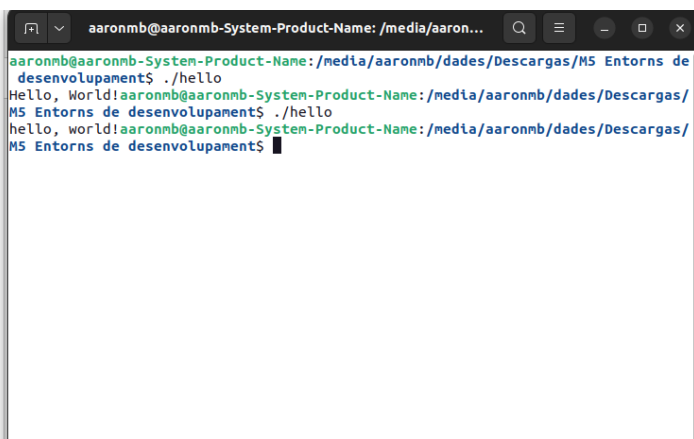


```
End of assembler dump.
(gdb) disas /r
Dump of assembler code for function main:
0x000055555555149 <+0>: f3 0f 1e fa    endbr64
0x00005555555514d <+4>: 55            push    %rbp
0x00005555555514e <+5>: 48 89 e5      mov     %rsp,%rbp
=> 0x000055555555151 <+8>: 48 8d 05 ac 0e 00 00    lea     0xac0e0000(%rip),%rax
# 0x555555556004
0x000055555555158 <+15>: 48 89 c7      mov     %rax,%rdi
0x00005555555515b <+18>: b8 00 00 00 00    mov     $0x0,%eax
0x000055555555160 <+23>: e8 eb fe ff ff    call   0x555555550500 <printf@plt>
t>
0x000055555555165 <+28>: b8 00 00 00 00    mov     $0x0,%eax
0x00005555555516a <+33>: 5d            pop     %rbp
0x00005555555516b <+34>: c3            ret
End of assembler dump.
(gdb)
```

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

Ara amb l'okteta (editor de fitxers) hem pogut modificar el nostre fitxer canviant les majúscules per les minúscules del Hello, World! El que hem fet ha sigut modificar el codi hexadecimal amb l'okteta, on hem modificat la columna de la dreta sobre el codi de l'esquerra.

```
E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
100 01 00 02 00 68 65 6C 6C 6F 2C 20 77 6F 72 6C 64 ...hello, world
110 21 00 00 00 01 18 03 3B 30 00 00 00 05 00 00 00 ...!.....;0.....
120 0C F0 FF FF 64 00 00 00 2C F0 FF FF 8C 00 00 00 ...öÿd...öÿ...
130 3C F0 FF FF A4 00 00 00 4C F0 FF FF 4C 00 00 00 ...<öÿ=...LöÿL...
140 35 F1 FF FF BC 00 00 00 14 00 00 00 00 00 00 00 ...5hÿÿ...
150 01 7A 52 00 01 78 10 01 1B 0C 07 08 00 01 00 00 ...zR..X.....
160 14 00 00 00 1C 00 00 00 F8 EF FF FF 26 00 00 00 ...øiÿÿ%...
170 00 44 07 10 00 00 00 00 24 00 00 00 34 00 00 00 ...D.....$.4...
180 A0 EF FF FF 20 00 00 00 00 0E 10 46 0E 18 4A 0F ...iÿÿ...F..J...
190 0B 77 08 80 00 3F 1A 3A 2A 33 24 22 00 00 00 00 ...w...?.:*3$"...
1A0 14 00 00 00 5C 00 00 00 98 EF FF FF 10 00 00 00 ...\\...iÿÿ...
1B0 00 00 00 00 00 00 00 00 14 00 00 00 74 00 00 00 ...t...
1C0 00 EF FF FF 10 00 00 00 00 00 00 00 00 00 00 00 ...iÿÿ...
1D0 1C 00 00 00 8C 00 00 00 71 F0 FF FF 23 00 00 00 ...qöÿÿ#...
1E0 00 45 0E 10 86 02 43 0D 06 5A 0C 07 08 00 00 00 ...E...C..Z...
1F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
210 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
220 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
230 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
240 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
250 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
260 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
270 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
```



```
aaronmb@aaronmb-System-Product-Name: /media/aaronmb/dades/Descargas/M5 Entorns de desenvolupament$ ./hello
Hello, World!aaronmb@aaronmb-System-Product-Name: /media/aaronmb/dades/Descargas/M5 Entorns de desenvolupament$ ./hello
hello, world!aaronmb@aaronmb-System-Product-Name: /media/aaronmb/dades/Descargas/M5 Entorns de desenvolupament$
```