Op logic	Formula echivalentă	Operație	formula echivalentă
A XOR B	(A+B) − 2(A AND B)	A+B	(A XOR B) + 2(A AND B)
A OR B	(A+B) − (A AND B)	A−B	(A XOR B) − 2((NOT A) AND B)
A AND B	$\frac{(A+B)-(A \text{ XOR } B)}{2}$	−A	(NOT A)+1
NOT A	(−A) − 1		

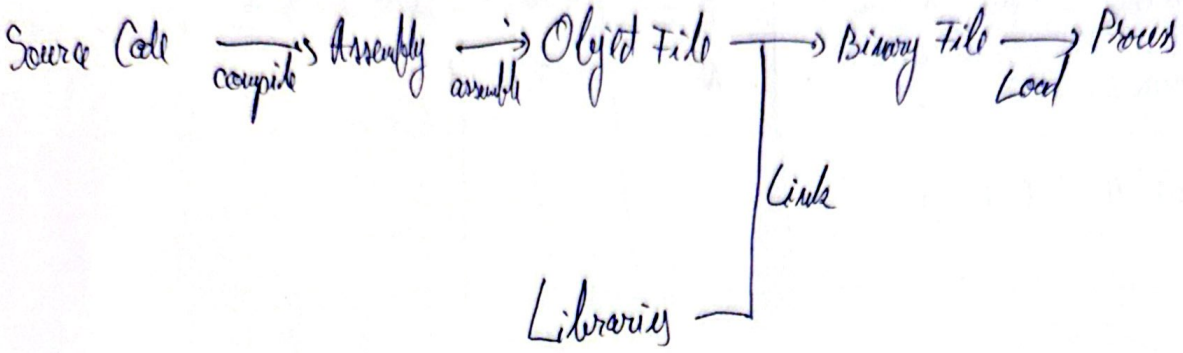
NOT	AND	OR	XOR
NOT A	A AND B	A OR B	A XOR B
$\overline{A}$	A×B	A+B	A⊕B
¬A	A•B	A∨B	A^B
~A	A*B	A | B	A != B
-A	A.B	A || B	
A'	AB		
!A	A∧B		
	A&B		
	A&&B		

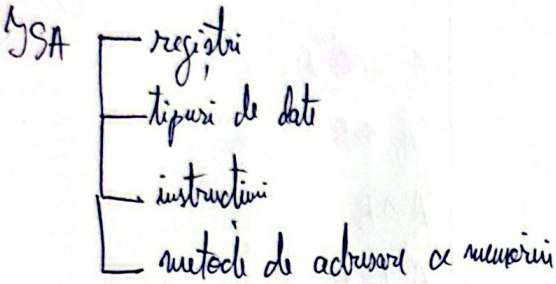
Bootloader: adresa 0x4A57 (găsit), ~ 0x7C00 (încărcat)

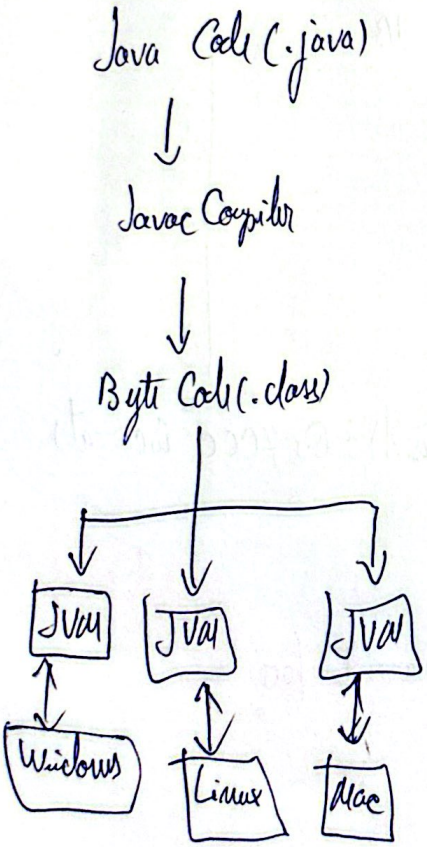
Hxd = hex editor

UEFI = Unified Extensible Firmware Interface

```
as --32 program_exit.asm -o program_exit.o
ld -m elf_i386 program_exit.o -o program_exit
```







Cod interpretat:

Java: Java byte-code → Java VM

C#: Common Intermediate Language (CIL) → Common Language Runtime (CLR) în .NET

Python: .py, python byte-code → python VM

Javascript: .js → V8 sau Spider Monkey

JIT (Just-In-Time Compilation)

op întreg / biți	1
înmulțirea întregilor	3
împărțirea întregilor	depinde (20-80)
adunare FP	3
înmulțire FP	5
împărțire FP	(depinde) 20-80
fused-multiply-add FP	5

Option	optim. level	Exec. time	Code size	mem. usage	Compile time
-O0	Optimization for compile time (default)	+	+	−	−
-O1 or -O	— II — for code size and execution time	−	−	+	+
-O2	— II — code size more for — II —	− −		+	+ +
-O3	— II —	− − −		+	+ + +
-Os	— II — code size		− −		+ +
-Ofast	O3 with fast and accurate math calculations	− − −		+	+ + +