

# Compilers

## Before and After Compilation

**Before:** Three higher-level languages

<pre> 10 A = 3 20 X = 1 30 print X 40 X = X + 1 50 IF X &lt;&gt; A THEN GOTO 30 60 PRINT "DONE" </pre>	<pre> int limit = 3; int val = 1; repeat {     console.write(val);     ++val; } until (val == limit) console.write("DONE"); </pre>	<pre> var limit = 3; var val = 1; do {     alert(val);     ++val; } while (val &lt;&gt; limit); alert("DONE"); </pre>
--	--	---

**During:** Intermediate Representation (6502a Assembly. Ms-IL and Java bytecodes are similar.)

lda #\$3	Acc = 3	0000	LDA #\$03	A9 03
sta \$0041	Mem[41] = 3	0002	STA \$0041	8D 41 00
lda #\$1	Acc = 1	0005	LDA #\$01	A9 01
sta \$0040	Mem[40] = 1	0007	STA \$0040	8D 40 00
loop ldy \$0040	Y = Mem[40]	000A <b>LOOP</b>	LDY \$0040	AC 40 00
ldx #\$01	X = 1	000D	LDX #\$01	A2 01
sys	System Call	000F	SYS	FF
inc \$0040	Mem[40]++	0010	INC \$0040	EE 40 00
ldx \$0040	X = Mem[40]	0013	LDX \$0040	AE 40 00
cpx \$0041	Z bit = (x == Mem[41])	0016	CPX \$0041	EC 41 00
bne loop	if z == 0 goto loop	0019	BNE <b>LOOP</b>	D0 EF
lda #\$44	Acc = \$44 ("D")	001B	LDA #\$44	A9 44
sta \$0042	Mem[42] = \$44	001D	STA \$0042	8D 42 00
lda #\$4F	Acc = \$4F ("O")	0020	LDA #\$4F	A9 4F
sta \$0043	Mem[43] = \$4F	0022	STA \$0043	8D 43 00
lda #\$4E	Acc = \$4E ("N")	0025	LDA #\$4E	A9 4E
sta \$0044	Mem[44] = \$4E	0027	STA \$0044	8D 44 00
lda #\$45	Acc = \$45 ("E")	002A	LDA #\$45	A9 45
sta \$0045	Mem[45] = \$45	002C	STA \$0045	8D 45 00
lda #\$00	Acc = \$00 (null)	002F	LDA #\$00	A9 00
sta \$0046	Mem[46] = \$00	0031	STA \$0046	8D 46 00
ldx #\$02	X = 2	0034	LDX #\$02	A2 02
ldy #\$42	Y = \$42 (address)	0036	LDY #\$42	A0 42
sys	System call	0038	SYS	FF
brk	Break	0039	BRK	00

**After:** Machine Language

A9 03 8D 41 00 A9 01 8D 40 00 AC 40 00 A2 01 FF EE 40 00 AE 40 00 EC 41 00 D0
EF A9 44 8D 42 00 A9 4F 8D 43 00 A9 4E 8D 44 00 A9 45 8D 45 00 A9 00 8D 46 00
A2 02 A0 42 FF 00