

```
b=5;
a=3;
c=7;
if (c>b+a) c=1
eles c=0
```

a	128	3
b	129	5
c	130	7
5	131	5
3	132	3
7	133	7
1	134	1
0	135	0

```
b=5;
a=3;
c=7;
if (c>b+a) c=1
else c=0
```

```
LDA 131
STA 129
LDA 132
STA 128
LDA 133
STA 130 AC = 7 → C
NOT
ADD 134 ; -C ou -7
ADD 129
ADD 128 ; -c +b +a
JN C_RECEBE_0
JZ C_RECEBE_0
LDA 134
STA 130 ; C = 1
```

JMP FIM

C_RECEBE_0:

LDA 135

STA 130;

FIM:

HLT

(...)

$c > b+a$

$0 > b + a - c$ (falhar JZ ou JN)

OU

$-c < -b -a$

$0 < c -b -a$ (JN for verdadeiro)

Voltando...

i=0;	i 128	0
while (i<4){	a 129	0
i++;	4 130	4
a = i + 5;	5 131	5
}	1 132	1

$AC < 4 \rightarrow i - 4 < 0$

WHILE:

LDA 130 AC = 4

NOT

ADD 132 ; AC = -4

ADD 128 ;

JN EXECUCAO_WHILE

JMP FIM

EXECUCAO_WHILE:

LDA 128;

ADD 132

STA 128;

LDA 128;

ADD 131;

STA 129;

JMP WHILE

FIM:

HLT

```

x = 3
while (x < 6){
    x ++;
    y = y + 2;
}

```

```

x 128 0
y 129 0
1 130 1
2 131 2
3 132 3
6 133 6

```

```

LDA 132
STA 128    ; x = 3

```

```

WHILE:
; x < 6    0 < 6 - x    →    0 > x-6

```

```

LDA 133 ; AC = 6 ; binario : 0 0 0 0 0 1 1 0
NOT;                      1 1 1 1 1 0 0 1
ADD 130 ; AC = -6;        1 1 1 1 1 0 1 0
ADD 128 ; AC = -6 + x
JN EXECUTO_WHILE
JMP FIM

```

```

EXECUTO_WHILE :
LDA 128;
ADD 130; AC = X + 1
STA 128 ; x++
LDA 129;
ADD 131; ac = y+ 2
STA 129 ; y = y + 2;

```

JMP WHILE

FIM:

HLT