1 inst 1 10

2 inst 2 15 20

3 escapar 30

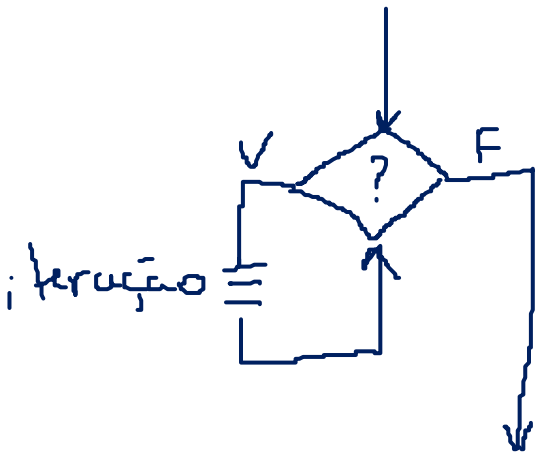
4 40

5 goto 1

256

Estrutura de Repetição

início
passo
fim



$-2^{31} \longrightarrow 2^{31} - 1$

MCINT

-2 -1 0 1 2

1 + 1 = 2

2 + 1 = -2

2 + 2 = -1

2 + 2 + 2 = 1

```
x = 1
enquanto x < 2
    x = x - 1
    print (x)
fim
```

x	0
x	-1
ø	-2
-1	2
-2	
2	

Acumuladores

var = var op expressão

pré
pós Incremento

cont = cont + 1 : cont++

pré
pós Decremento

i = i - 1 : i--

Exemplos

cont = cont + 1;

i = i - 1;

m = m * 10;

d = d % 2;

K = K / (a+b);

msg = msg + "oi";

i-- = -i

m *= 10

d %= 2

cont = cont++;

K += (a+b)

```
m = 0.02
while (m <= 0.75) {
    m = m + 0.1;
}
```

0.02
m++
1.02

```
int i = 10;
printf ("%d", i++);
```

Mem
i 10 11

tela
10

```
int= 10;
printf ("%d", ++i);
```

mem
i 10 11

tela
11

```
int i = 10;
printf ("%d", i++);
```

mem
i 10

tela
11

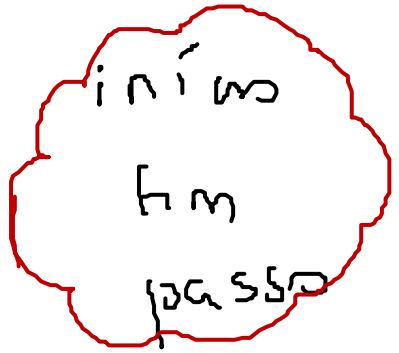
```
int n=1;
while (n<=4) {
    pf (n); // proc
    n++;    //at
}
printf ("%d", n);
```

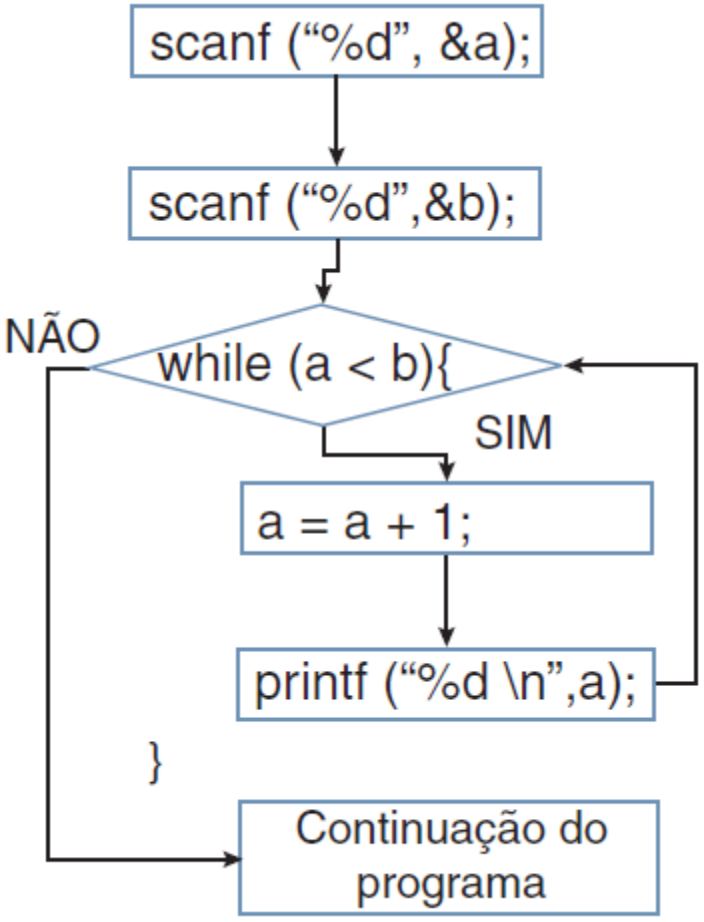
Mem	Tela
~~1~~	1
~~2~~	2
~~3~~	3
~~4~~	4
5	↑

n

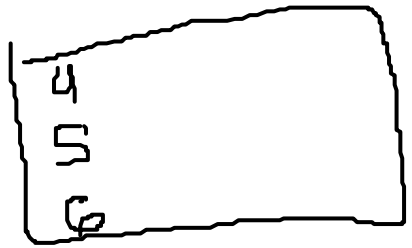
```
n=1;
while (n<=4) {
    n++;    // at
    pf (n); // proc
}
->printf ("%d", n);
```

M	T
~~1~~	2
~~2~~	3
~~3~~	4
~~4~~	5
5	↑





a = 3 4 5 6
b = 6




```
scanf ("%d%d", &a, &b);
   while (a < b) {
      printf ("%d\n", a);
      a++;
   }
```

a = 3 4 5 6
b = 6

```
scanf ("%d", &n);
cont = 1; // 0
soma = 0;
while (cont <= n) {
    scanf ("%d", &val);
    soma = soma + val;
    cont++;
}
media = soma / n;
```

Rascunho

$m = (v1 + v2 + v3) / 3$

propriedades

elemento neutro
- soma = 0
- multiplicação = 1

Teste

início

for

while

Fim E do - while

Controle

contador

lógico
(sentinela)

```
i = 3;
while(i < 2){
    printf("*");
    i++;
}
```