

myproc(j:int, par:int)

pass by value

a[1]=0, a[2]=0, a[3]=0

a[2]=2

j=1

myproc(1, 0)

inside myproc

a[1]=3

j=0

par = a[0] + 1

writeln(j, a[1], a[2], a[3])

↓	↓	↓	↓
1	3	2	0

myproc(j:int, par:int)

a[1] = 0, a[2] = 0, a[3] = 0

a[2] = 2

j = 1

myproc(j, a[1])

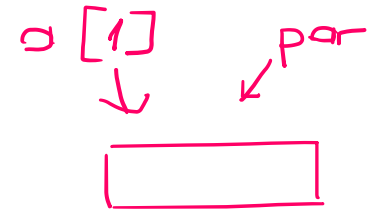
inside myproc

a[1] = 3

j = a[1] = 3

par = a[3] + 1 = 1

pass by ref.



writeln(j, a[1], a[2], a[3])

↓	↓	↓	↓
3	1	2	0

pass by value result

myproc(j: int, par: int)

a[1] = 0, a[2] = 0, a[3] = 0

a[2] = 2

j = 1

myproc(1, 0)

inside myproc

a[1] = 3

j = 0

par = a[0] + 1

copy in  
copy out!

writeln(j, a[1], a[2], a[3])

↓  
0

↓  
a[0] + 1

↓  
2

↓  
0

myproc(j:int, par:int)

a[1]=0, a[2]=0, a[3]=0

a[2]=2

j=1

myproc(j, a[j])

inside myproc

a[1]=3

j = a[1] = 3

par = a[3] + 1 = 1

pass by name

println(j, a[1], a[2], a[3])  
3 3 2 1

Each use of the formal reevaluates the actual in the caller's environment.