```c
void para_par (int a[], int n) {
   int i;

   for (i=0 ; i<n ; i++) {
      if (a[i] & 0x01) {
         a[i] += 1;
      }
   }
}
```

---- código assembly obtido com gcc -S -O0 ----

```
...
para_par:
        pushl   %ebp
        movl    %esp, %ebp
        pushl   %ebx
        subl    $4, %esp
        movl    $0, -8(%ebp)
.L2:
        movl    -8(%ebp), %eax
        cmpl    12(%ebp), %eax
        jl      .L5
        jmp     .L1
.L5:
        movl    -8(%ebp), %eax
        leal    0(,%eax,4), %edx
        movl    8(%ebp), %eax
        movl    (%eax,%edx), %eax
        testl   $1, %eax
        je      .L4
        movl    -8(%ebp), %eax
        leal    0(,%eax,4), %ebx
        movl    8(%ebp), %ecx
        movl    -8(%ebp), %eax
        leal    0(,%eax,4), %edx
        movl    8(%ebp), %eax
        movl    (%eax,%edx), %eax
        incl    %eax
        movl    %eax, (%ecx,%ebx)
.L4:
        leal    -8(%ebp), %eax
        incl    (%eax)
        jmp     .L2
.L1:
        addl    $4, %esp
        popl    %ebx
        leave
        ret
```

---- código assembly obtido com gcc -S -O2 ----

```
...
para_par:                                    %eax — índice
        pushl   %ebp
        movl    %esp, %ebp                   %ebx — ?
        xorl    %edx, %edx
        pushl   %ebx                         %ecx — início
        movl    12(%ebp), %ebx
        cmpl    %ebx, %edx                   %edx — n
        movl    8(%ebp), %ecx
        jge     .L9
.L7:
        movl    (%ecx,%edx,4), %eax
        testl   $1, %eax
        je      .L4
        incl    %eax
        movl    %eax, (%ecx,%edx,4)   onho to i}
.L4:
        incl    %edx
        cmpl    %ebx, %edx
        jl      .L7
.L9:
        popl    %ebx
        leave
        ret
```

---- executável com -O2 após objdump -d -----

```
...
08048354 <para_par>:
 8048354:  55              push   %ebp
 8048355:  89 e5           mov    %esp,%ebp
 8048357:  31 d2           xor    %edx,%edx
 8048359:  53              push   %ebx
 804835a:  8b 5d 0c        mov    0xc(%ebp),%ebx
 804835d:  39 da           cmp    %ebx,%edx
 804835f:  8b 4d 08        mov    0x8(%ebp),%ecx
 8048362:  7d 13           jge    8048377
 8048364:  8b 04 91        mov    (%ecx,%edx,4),%eax
 8048367:  a9 01 00 00 00  test   $0x1,%eax
 804836c:  74 04           je     8048372
 804836e:  40              inc    %eax
 804836f:  89 04 91        mov    %eax,(%ecx,%edx,4)
 8048372:  42              inc    %edx
 8048373:  39 da           cmp    %ebx,%edx
 8048375:  7c ??           jl     8048364
 8048377:  5b              pop    %ebx
 8048378:  c9              leave
 8048379:  c3              ret
```

- breakpoint em para_par (executável com -O2)

```
(gdb) info registers
eax        0x2              2
ecx        0x8049620
edx        0x2              2
ebx        0x14             20
esp        0xbfffe8a4
ebp        0xbfffe8a8
eip        0x8048364

--------------------------

(gdb) x/5wx $esp
0xbfffe8a4:
   0x007d3ff4  0xbfffe8c8  0x080483a4  0x08049620
0xbfffe8b4:
   0x00000014

--------------------------

(gdb) x/20wb $ecx
0x08049620:
   0x00000000 0x00000002 0x00000002 0x00000003
0x08049630:
   0x00000004 0x00000005 0x00000006 0x00000007
0x08049640:
   0x00000008 0x00000009 0x0000000a 0x0000000b
0x08049650:
   0x0000000c 0x0000000d 0x0000000e 0x0000000f
0x08049660:
   0x00000010 0x00000011 0x00000012 0x00000013
```