

## Líneas afectadas por los cambios

### VERSIÓN 1 :Evalúa las guardas del bucle arriba

#### :Condiciones del bucle

```
while CMP R3,#1      ;Comprueba si la posición es válida  
    BNE if  
    CMP R4,R9      ;Comprueba si la casilla es igual al color  
    BEQ if
```

#### :Cuerpo del bucle

```
bucle ADD R2,R5,R7      ;FA = FA + SF  
    MOV R2,R2,LSL #24  ;Conserva solo los 8 primeros bits  
    MOV R5,R2,ASR #24  ;Expande el número con signo de 8 bits, a uno de 32  
    ADD R2,R6,R8      ;CA = CA + SC  
    MOV R2,R2,LSL #24  ;Conserva solo los 8 primeros bits  
    MOV R6,R2,ASR #24  ;Expande el número con signo de 8 bits, a uno de 32  
  
    LDR R2,[R1]  
    ADD R2,R2,#1      ;Longitud = Longitud + 1;  
    STR R2,[R1]       ;Actualiza longitud en memoria
```

;ficha\_valida Se omite porque no es relevante para esta versión

#### B while

### VERSIÓN 2 : Evalúa las condiciones abajo

#### :Cuerpo del bucle

```
bucle ADD R2,R5,R7      ;FA = FA + SF  
    MOV R2,R2,LSL #24  ;Conserva solo los 8 primeros bits  
    MOV R5,R2,ASR #24  ;Expande el número con signo de 8 bits, a uno de 32  
    ADD R2,R6,R8      ;CA = CA + SC  
    MOV R2,R2,LSL #24  ;Conserva solo los 8 primeros bits  
    MOV R6,R2,ASR #24  ;Expande el número con signo de 8 bits, a uno de 32  
  
    LDR R2,[R1]  
    ADD R2,R2,#1      ;Longitud = Longitud + 1;  
    STR R2,[R1]       ;Actualiza longitud en memoria
```

;ficha\_valida Se omite porque no es relevante para esta versión

#### :Condiciones del bucle

```
while CMP R3,#1      ;Comprueba si la posición es válida  
    BNE if  
    CMP R4,R9      ;Comprueba si la casilla es igual al color  
    BNE bucle
```

