

En un procesador VLIW con 3 slots todas las operaciones pueden predicarse. Para establecer los valores de los predicados se utilizan instrucciones de comparación con el formato $\rightarrow (p) p1, p2 \text{ cmp } cnd \ x, y$ donde cnd es la condición. Si se cumple la condición $p1 = 1$ y $p2 = 0$ y viceversa en caso contrario. Escribir el siguiente código desenrollando el bucle, sin ninguna instrucción de salto, con el mínimo número de instrucciones VLIW.

```
for (i=1; i<=5; i++){
    if (v[i]<0)
        v[i] = -v[i];
    else
        v[i]=v[i]+2;
}
```

Inicialización de predicado p1:
 p1 cmp.eq r0, r0

Instrucciones de carga:

```
(p1)  ld r2, 0(r1)
(p1)  ld r3, 8(r1)
(p1)  ld r4, 16(r1)
(p1)  ld r5, 24(r1)
(p1)  ld r6, 32(r1)
```

Comparaciones:

```
(p1)  p2,p3  cmp.lt r2, r0
(p1)  p4,p5  cmp.lt r3, r0
(p1)  p6,p7  cmp.lt r4, r0
(p1)  p8,p9  cmp.lt r5, r0
(p1)  p10,p11 cmp.lt r6, r0
```

Se cumple la condición:

```
(p2)  sub r2, r0, r2
(p4)  sub r3, r0, r3
(p6)  sub r4, r0, r4
(p8)  sub r5, r0, r5
(p10) sub r6, r0, r6
```

No se cumple la condición:

```
(p3)  addi r2, r2, #2
(p5)  addi r3, r3, #2
(p7)  addi r4, r4, #2
(p9)  addi r5, r5, #2
(p11) addi r6, r6, #2
```

Almacenamiento:

```
(p1) sw r2, 0(r1)      (p1) sw r3, 8(r1)      (p1) sw r4, 16(r1)
(p1) sw r5, 24(r1)     (p1) sw r6, 32(r1)     (p1) addi r1, r1, #40
```

SLOT 1	SLOT 2	SLOT 3
p1 cmp.eq r0, r0	---	---
(p1) ld r2, 0(r1)	(p1) ld r3, 8(r1)	(p1) ld r4, 16(r1)
(p1) ld r5, 24(r1)	(p1) ld r6, 32(r1)	(p1) p2,p3 cmp.lt r2, r0
(p1) p4,p5 cmp.lt r3, r0	(p1) p6,p7 cmp.lt r4, r0	(p1) p8,p9 cmp.lt r5, r0
(p1) p10,p11 cmp.lt r6, r0	(p2) sub r2, r0, r2	(p4) sub r3, r0, r3
(p6) sub r4, r0, r4	(p8) sub r5, r0, r5	(p10) sub r6, r0, r6
(p3) addi r2, r2, #2	(p5) addi r3, r3, #2	(p7) addi r4, r4, #2
(p9) addi r5, r5, #2	(p11) addi r6, r6, #2	(p1) sw r2, 0(r1)
(p1) sw r3, 8(r1)	(p1) sw r4, 16(r1)	(p1) sw r5, 24(r1)
(p1) sw r6, 32(r1)	(p1) addi r1, r1, #40	---