

Ejercicio 5:

Dados los siguientes programas en LEGv8:

5.1) ¿Cuántas instrucciones LEGv8 ejecuta cada uno?

5.2) Reescribir en “C” dada la asignación  $X10, X1, X2, X0 \leftrightarrow i, a, result, MemArray$ .

```

    ADD X10, XZR, XZR           // X10 = 0
loop: LDUR X1, [X0, #0]         // X1 = &X0[0]
    ADD X2, X2, X1              // X2 = X2 + &X0[0]
    ADDI X0, X0, #8             // &X0[0] = &X0[0] + 8 = &X0[1]
    ADDI X10, X10, #1           // X10 = X10 + 1
    CMPI X10, #100             // FLAGS = X10 - 100
    B.LT loop                   // si X10 < 100, salto a loop
done:
```

Éste bloque de código ejecuta 501 instrucciones LEGv8 (100 veces un loop de 5 instrucciones y la inicialización de X10)

```

En C:
i = 0;
while (i < 100) {
    a = MemArray[0];
    result += a;
    MemArray = MemArray + 8;
    i++;
}
```

```

de igual manera:
for (long i=0; i < 100; i++) {
    a = MemArray[0];
    result += a;
    MemArray += 8;
}
```

```

    ADDI X10, XZR, #50          // X10 = 50
loop: LDUR X1, [X0, #0]         // X1 = &X0[0]
    ADD X2, X2, X1              // X2 = X2 + &X0[0]
    LDUR X1, [X0, #8]           // X1 = &X0[1]
    ADD X2, X2, X1              // X2 = X2 + &X0[1]
    ADDI X0, X0, #16            // X0 = X0 + 16
    SUBI X10, X10, #1           // X10 = X10 - 1
    CNBZ X10, loop              // si X10 ≠ 0, salto a loop
done:
```

Éste bloque de código ejecuta 351 instrucciones LEGv8 (50 veces un loop de 7 instrucciones y la inicialización de X10)

```

En C:
i = 50;
while (i != 0) {
    a = MemArray[0];
    result += a;
    a = MemArray[1];
    result += a;
    MemArray += 16;
    i--;
}
```

```

de igual manera:
for (long i=50; i != 0; i--) {
    a = MemArray[0];
    result += a;
    a = MemArray[1];
    result += a;
    MemArray += 16;
}
```

```

reduciendo líneas de código:
for (long i=50; i != 0; i--) {
    a = MemArray[0];
    result += a;
    MemArray += 8;
}
```

```

en assembler:
    ADDI X10, XZR, #50          // X10 = 50
loop: LDUR X1, [X0, #0]         // X1 = &X0[0]
    ADD X2, X2, X1              // X2 = X2 + &X0[0]
    ADDI X0, X0, #8             // X0 = &X0[0] + 8 = &X[1]
    SUBIS X10, X10, #1          // FLAGS = X10 - 1
    B.GE loop                   // si X10 ≥ 1 salto a loop
done:
```