Ejercicio 01

Inciso a: dadd r1, r2, r0

Esta instrucción tiene 3 operandos que son de tipo **double word**. Aunque para el MIPS una palabra sean 64 bits, muchas instrucciones fueron reutilizadas de la versión anterior donde valía 32 bits. ~~(Que vagancia la de Francia).~~

Al ejecutarse, se suman el contenido de los registros r2 y r0, siendo éste último 0. El resultado (r2 + 0 = r2) es asignado en el registro r1. Por lo tanto: **r1 = r2**

---

*En MSX88, su equivalente sería **MOV R1, R2***

Inciso b: daddi r3, r0, 5

Ídem anterior, pero los sumandos son un registro (r0 = 0) y un valor inmediato (5). El resultado (0 + 5 = 5) es asignado en el registro r3. Por lo tanto: **r3 = 5**

---

*En MSX88, su equivalente sería **MOV R3, 5***

Inciso c: dsub r4, r4, r4

Ídem inciso a, pero se resta el contenido del registro r4 por éste mismo valor. El resultado (r4 – r4 = 0) es asignado en el registro r4. Por lo tanto: **r4 = 0**

---

*En MSX88, hay 2 equivalentes: **MOV R4, 0** **SUB R4, R4***

Inciso d: daddi r5, r5, -1

Ídem inciso b, pero el operando fuente de tipo registro no es cero (r5). El resultado (r5 + (-1)) es asignado en el registro r5. Por lo tanto: **r5 -= 1**

---

*En MSX88: **DEC R5** **SUB R5, 1** **ADD R5, -1***

Inciso e: xori r6, r6, 0xffffffffffffffff

Ídem anterior, pero se realiza una operación lógica (máscara XOR). El resultado (invertir los 64 bits de r6) es asignado en el registro r6.

---

*En MSX88: **NOT R6** **XOR R6, 0FFFFFFFFFFFFFFFh***
