

Ejercicio 1	Análisis de programa en lenguaje de ensamble	1.8 puntos
-------------	--	------------

La memoria RAM de un microcontrolador basado en un procesador ARM Cortex-M0 (little-endian) contiene los datos dados en la siguiente tabla:

Dirección / y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x1000 000y	2F	A7	30	3D	2E	7C	43	12	FF	FF	C6	24	F8	FF	3D	35
0x1000 001y	7F	73	0F	F0	97	6C	E5	4B	6A	2A	A4	75	D1	0C	33	65
0x1000 002y	00	00	8A	6A	49	2C	75	3C	C3	EA	3B	3E	F8	36	DF	07

Los valores de algunos registros del procesador son:

Registro	Contenido
R0	0x1000 0002
R1	0xDEAD BEEF
R2	0xFEE1 BAAD
R3	0xA5A5 5A5A

En estas circunstancias se llama a una función cuyo prototipo es `int fun(char *)` y cuyo cuerpo es:

```
fun
MOV R1, R0
MOVS R0, #0
MOV R2, R0
SUBS R2, #2
fun_loop
ADDS R2, #2
LDRSH R3, [R1, R2]
CMP R3, #0
BEQ fun_exit
BPL fun_loop
RSBS R3, R3, #0
STRH R3, [R1, R2]
ADDS R0, #1
B fun_loop
fun_exit
BX LR ; return
```

$R_1 = 0x1000\ 0002$
 $R_0 = 0$ ($z == 1$)
 $R_2 = 0$
 $R_3 = 0x0000\ 3D30$

Apartado A. Rellene la siguiente tabla indicando cómo quedarán los registros del procesador tras la ejecución de `fun()` (es decir, cuando se alcance la instrucción `BX LR`, que retorna de la función).

Registro	Contenido (hexadecimal con 32 bits)
R0	00... 3
R1	0x1000 0002
R2	0x0... 001E
R3	0