







3) Considere os endereços de início dos segmentos: `.data = 0x10010000` e `.text = 0x00400000`. Interprete o trecho de código *Assembly* do MIPS apresentado.

Linha	Label	Assembly	Comentário
1		<code>.data</code>	
2	<code>oper1:</code>	<code>.word -1</code>	
3	<code>oper2:</code>	<code>.byte 3</code>	
4		<code>.align 2</code>	
5	<code>result:</code>	<code>.space 4</code>	
6		<code>.text</code>	
7	<code>main:</code>	<code>la \$t0, oper1</code>	
8		<code>la \$t1, oper2</code>	
9		<code>lb \$t2, 0(\$t1)</code>	
10			
11		<code>lw \$t3, 0(\$t0)</code>	
12	<code>loop:</code>	<code>beq \$t2, \$0, endl</code>	
13		<code>sll \$t3, \$t3, 1</code>	
14		<code>addi \$t2, \$t2, -1</code>	
15		<code>j loop</code>	
16	<code>endl:</code>	<code>la \$t5, result</code>	
17		<code>sw \$t3, 0(\$t5)</code>	
18		<code>add \$a0, \$t3, \$0</code>	
19		<code>li \$v0, 34</code>	
20		<code>syscall</code>	<code>#PRINT_INT16 (unsigned)</code>
21		<code>li \$v0, 10</code>	
22		<code>syscall</code>	<code>#EXIT</code>
23			

a) Na linha 16 (`la $t5, result`) quais as instruções nativas geradas?

Para cada uma dessas instruções nativas, indique a **mnemónica Assembly**, o **registo destino** e o **valor das constantes**.

b) Na linha 9 (`lb $t2, 0($t1)`) qual o valor armazenado em **\$t2**?

E na linha 11 (`lw $t3, 0($t0)`) qual o valor armazenado em **\$t3**?

\$t2 = \_\_\_\_\_ \$t3 = \_\_\_\_\_

c) Quantas vezes é executada a instrução da linha 13 (`sll $t3, $t3, 1`)? Qual o valor final de **\$t3**?

Se não respondeu à alínea b) considere o valor de **\$t2 = 0x04**.

d) Qual o resultado impresso no ecrã pela `syscall` da linha 20?

Se não respondeu à alínea c) considere o valor de **\$a0 = -4**.