

2 Question 1) Convert the following C code to MIPS. Assume the variable 'result' will correspond to register \$s0

```
void main()
{
    int s1 = 18;
    int s2 = 6;
    int result = s2 + s1 - 13;
}
```

Answer here:

```
addi $s1, $zero, 18
addi $s2, $zero, 6
add $s0, $s2, $s1
addi $s0, $s0, -13
```

1.5 Question 2) Given that the array of integers (4 bytes each int) 'my_array' contains the following elements [19, 9, 8, 8, 10]; also, assume that the register \$s0 points to (or contains the address of) the first element in the array. List the values of 'my_array' after executing the code below:

lw \$t4, 4(\$s0)	t4 = 9
lw \$s6, 16(\$s0)	s6 = 10
sub \$t5, \$s6, \$t4	t5 = 10 - 9 = 1
sub \$t2, \$t5, \$s6	t2 = 1 - 10 = -9
add \$s1, \$t5, \$s6	s1 = 1 + 10 = 11
sw \$t2, 16(\$s0)	arr[4] = -9
sw \$s1, 0(\$s0)	arr[0] = 11
sw \$t4, 12(\$s0)	arr[3] = 9
sw \$t2, 0(\$s0)	arr[0] = -9
sw \$s6, 12(\$s0)	arr[3] = 10

Final array: [-9, 9, 8, 10, -9]

2.5 Question 3) Given that the array of integers (4 bytes each int) 'my_array' contains the following elements [12, 11, 19, 12, 12]; also, assume that the register \$s0 points to (or contains the address of) the first element in the array. List the values of these registers \$s1, \$s6, \$t0, \$t4, \$t5 in the given order after executing the code below:

lw \$t5, 8(\$s0)	t5 = 19
addi \$s6, \$zero, 8	s6 = 8
sub \$s1, \$s6, \$s6	s1 = 0
lw \$t0, 12(\$s0)	t0 = 12
sw \$s1, 12(\$s0)	arr[3] = 0
add \$t4, \$t0, \$t5	t4 = 19 + 12 = 31

Question 4) How many bytes in a word?

4

Question 6) What's the name of the component that translates Assembly to Machine Language?

Assembler

Question 7) What's the name of the component that translates C to Assembly?

Compiler

Question 8) In the context of registers, what does PC stand for?

Program Counter

Question 9) How many bits in a half word?

16

Question 10) T/F

- Accessing registers is as fast as accessing the random access memory (RAM).

False

- Assembly language is a low-level programming language

True

- In modern computers, memory is bit-addressable; in other words, one particular address refers to one particular bit.

False

- The high-level programming language C is architecture-dependent.

False

- In MIPS-32 (32-bit MIPS), each pure instruction will be assembled to 32 bits.

True