Q.1 Provide the instruction type, assembly language instruction, and binary representation of the instruction described by the following MIPS fields:

op = 0, rs = 3, rt = 2, rd = 3, shamt = 0, funct = 34

Solution:

R-type:

sub $v1, $v1, $v0

0x00621822

Q.2 Assume that we would like to expand the MIPS register file to 128 registers and expand the instruction set to contain four times as many instructions.

(a) How would this affect the size of each of the bit fields in the R-type instructions?
(b) How could each of the two proposed changes decrease the size of an MIPS assembly program? On the other hand, how could the proposed change increase the size of an MIPS assembly program?

Solution:

(a) The opcode or function code would expand from 6 bits to 8. The rs1, rs2 , and rd fields would increase from 5 bits to 7 bits.

(b) Increasing the size of each bit field potentially makes each instruction longer, potentially increasing the code size overall. * However, increasing the number of registers could lead to less register spillage, which would reduce the total number of instructions, possibly reducing the code size overall.

Q.3 $t0 = 0xAAAAAAAA, $t1 = 0x12345678

For the register values shown above, what is the value of $t2 for the following sequence of instructions?

sll $t2, $t0, 4
or $t2, $t2, $t1

Solution:

0xBABEFEF8

Q.4 For the following C statement, write a minimal sequence of MIPS assembly instructions that does the identical operation. Assume $t0 = A, and $s0 is the base address of C.

```
A = C[0] << 4;
```

Solution:

```
lw $t3, 0($s0)
sll $t1, $t3, 4
```