This question deals with the encoding of the conditonal branch instructions. Assume that the text segment starts at 0x00400000 and the data segments starts at 0x10010000. The implementation we are considering is the GT MIPS datapath. Assume each instruction can be encoded in one word.

A.

```
            .data
L1:         .word 0x32, 104
            .asciiz “Test 1”
            .align 2
Blank:      .word
L2:         .space 64

            .text
main:       li $t0, 4
            move $t2, $zero
            li $a0, 1
loop:       jal solo                # this is a procedure call
            addi $t0, $t0, -1
            addi $a0, $a0, 1
            add $t2, $t2, $v0
            slt $t1, $t0, $zero
            bne $t1, $zero, loop
            li $v0, 10
            syscall
```

What is the encoding of the following instruction ?

bne $t1, $zero, loop ____0x1520fffb_____

B. .

```
            .data
            .word 24, 0x16
L2:         .byte, 77,66,55,44
            .text
            move $t0,$0
loop:       mul $t8, $t0, 4
            add $t1, $a0, $t8
            sw $0, 0($t1)
            addi $t0, $t0, 1
            slt $t7, $t0, $a1
            bne $t7, $0, loop
```

Provide the hexadecimal encoding of the following instruction.

bne $t7, $0, loop 0x15e0fffb

.C.

```
            .data
label:  .word 8,16,32,64
            .byte 64, 32

            .text
            .globl main
main:       la $4, label
            li $5, 16

func:       move $2, $4
            move $3, $5
            add $3, $3, $2
            move $9, $0
loop:       lw $22, 0($2)
            add $9, $9, $22
            addi $2, $2, 4
            slt $8, $2, $3
            bne $8, $0, loop
            move $2, $9
            addi $v0, $0, 10
            syscall
```

What is the hexadecimal encoding of the following instruction.

bne $8, $0, loop 0x1500fffc