EXERCISE

1. Modify the given code to perform Vector Subtraction (A[i] - B[i]).

SOURCE CODE

```
#CODE BY TAMIA NAEEM CTAI-004
.data
A:.word 1, 3, 14, 9, 11
B:.word 33, 22, 33, 7, 4
C:.space 20
size:.word 5
spacing: .asciiz " "

.text
main:
    la $t0, A
    la $t1, B
    la $t2, C
    lw $t3, size
    li $t4, 0

loop:
    bge $t4, $t3, print_result
    mul $t5, $t4, 4
    add $t6, $t0, $t5
    add $t7, $t1, $t5
    lw $t8, 0($t6)
```

```
    lw $t9, 0($t7)
    sub $s0, $t8, $t9
    add $t6, $t2, $t5
    sw $s0, 0($t6)
    addi $t4, $t4, 1
    j loop

print_result:
    li $t4, 0

print_loop:
    bge $t4, $t3, end
    mul $t5, $t4, 4
    add $t6, $t2, $t5
    lw $a0, 0($t6)
    li $v0, 1
    syscall
    li $v0, 4
    la $a0, spacing
    syscall
    addi $t4, $t4, 1
    j print_loop
```

```
end:
    li $v0, 10
    syscall
```

OUTPUT

```
-32 -19 -19 2 7
-- program is finished running --
```

2. Implement Vector Multiplication of two arrays.

SOURCE CODE

```
#CODE BY TAMIA NAEEM CTAI-004
.data
A:      .word 6,7,82,3,3
B:      .word 2,3,2,1,5
C:      .space 20
size:   .word 5
spacing:.asciiz " "

.text
main:
    la $t0, A
    la $t1, B
    la $t2, C
    lw $t3, size
    li $t4, 0

loop:
    bge $t4, $t3, print_result
    mul $t5, $t4, 4
    add $t6, $t0, $t5
    add $t7, $t1, $t5
    add $t8, $t2, $t5
```

```
    lw $t6, 0($t6)
    lw $t7, 0($t7)
    mul $t9, $t6, $t7
    sw $t9, 0($t8)
    addi $t4, $t4, 1
    j loop

print_result:
    li $t4, 0

print_loop:
    bge $t4, $t3, end
    mul $t5, $t4, 4
    add $t6, $t2, $t5
    lw $a0, 0($t6)
    li $v0, 1
    syscall
    li $v0, 4
    la $a0, spacing
    syscall
    addi $t4, $t4, 1
    j print_loop

end:
    li $v0, 10
    syscall
syscall
```

OUTPUT

```
12 21 164 3 15
-- program is finished running --
```

3. Change the program to handle 10 elements instead of 5.

SOURCE CODE

```
#CODE BY TAMIA NAEEM CTAI-004
.data
A:        .word 1,3,5,7,9,1,13,2,9,2
B:        .word 2,4,6,3,14,15,16,3,3,10
C:        .space 40
size:     .word 10
spacing:  .asciiz " "

.text
main:
    la   $t0, A
    la   $t1, B
    la   $t2, C
    lw   $t3, size
    li   $t4, 0

compute_loop:
    bge  $t4, $t3, print_result
    mul  $t5, $t4, 4
    add  $t6, $t0, $t5
    add  $t7, $t1, $t5
    add  $t8, $t2, $t5
    lw   $t9, 0($t6)
    lw   $s0, 0($t7)
    mul  $s1, $t9, $s0
    sw   $s1, 0($t8)
    addi $t4, $t4, 1
```

```
28      j     compute_loop
29
30  print_result:
31      li    $t4, 0
32
33  print_loop:
34      bge   $t4, $t3, end
35      mul   $t5, $t4, 4
36      add   $t6, $t2, $t5
37      lw    $a0, 0($t6)
38      li    $v0, 1
39      syscall
40      li    $v0, 4
41      la    $a0, spacing
42      syscall
43      addi $t4, $t4, 1
44      j     print_loop
45
46  end:
47      li    $v0, 10
48      syscall
```

OUTPUT

```
ear
2 12 30 21 126 15 208 6 27 20
-- program is finished running --
```

4. Implement Scalar Multiplication: Multiply each element of A by a constant value (e.g.,

SOURCE CODE

```
#CODE BY TAMIA NAEEM CTAI-004
.data
A: .word 12,52,4,89,2
constant_value: .word 4
size: .word 5
spacing: .asciiz " "

.text
main:
    la $t0, A
    lw $t1, constant_value
    lw $t2, size
    li $t3, 0

loop:
    bge $t3, $t2, print_result
    mul $t4, $t3, 4
    add $t5, $t0, $t4
    lw $t6, 0($t5)
    mul $t6, $t6, $t1
    sw $t6, 0($t5)
    addi $t3, $t3, 1
    j loop

print_result:
    li $t3, 0
```

```
print_loop:
    bge $t3, $t2, end
    mul $t4, $t3, 4
    add $t5, $t0, $t4
    lw $a0, 0($t5)
    li $v0, 1
    syscall
    li $v0, 4
    la $a0, spacing
    syscall
    addi $t3, $t3, 1
    j print_loop

end:
    li $v0, 10
    syscall
```

OUTPUT

```
48 208 16 356 8
-- program is finished running --
```

5. Extend the code to calculate the dot product of two vectors.

SOURCE CODE

```
#CODE BY TAMIA NAEEM CTAI-004
.data
A: .word 9,8,72
B: .word 7,8,22
size: .word 3
dot_product: .word 0

.text
main:
    la $t0, A
    la $t1, B
    lw $t2, size
    li $t3, 0
    lw $t4, dot_product

loop:
    bge $t3, $t2, print_result
    mul $t5, $t3, 4
    add $t6, $t0, $t5
    add $t7, $t1, $t5
    lw $t6, 0($t6)
    lw $t7, 0($t7)
    mul $t8, $t6, $t7
    add $t4, $t4, $t8
    addi $t3, $t3, 1
    j loop
```

```
print_result:
    li $v0, 1
    move $a0, $t4
    syscall

end:
    li $v0, 10
    syscall
```

OUTPUT

```
1711
-- program is finished running --
```