

Computer Organization

Assignment 1: Sequential Construct-I

Sambit Sahoo (22277)

Q1 - compute the eight-term (in A.P) / fourth term (in G.P.) and sum of the first six (in A.P) / first four numbers (in G.P.).

```
#Sambit Sahoo 22277
#nth term AP =      a+(n-1)d
#sum upto n terms AP = n(2a+(n-1)d)/2
#nth term GP =      ar^(n-1)
#sum upto n terms GP = a(r^n-1)/(r-1)

.data

ap_a: .byte 1
ap_d: .byte 10
gp_a: .byte 4
gp_r: .byte 2

.text
.globl main2
main:
    #AP
    lb $t0, ap_a           #t0 = a
    li $t1, 8              #t1 = n
    addi $t1, $t1, -1      #t1 = n-1
    lb $t2, ap_d           #t2 = d
    mul $t3, $t2, $t1      #t3 = (n-1)d
    add $s0, $t0, $t3      #s0 = a+(n-1)d

    li $t0, 2              #t0 = 2
    lb $t1, ap_a           #t1 = a
    mul $t2, $t1, $t0      #t2 = 2a
    li $t3, 6              #t3 = n
    addi $t4, $t3, -1      #t4 = n-1
    lb $t5, ap_d           #t5 = d
    mul $t6, $t5, $t4      #t6 = (n-1)d
    add $t7, $t2, $t6      #t7 = 2a+(n-1)d
    mul $t8, $t3, $t7      #t8 = n(2a+(n-1)d)
    div $t8, $t0           #t8 = n(2a+(n-1)d)/2
    mflo $s1               #s1 = n(2a+(n-1)d)/2

    #GP
    lb $t0, gp_a           #t0 = a
    lb $t1, gp_r           #t1 = r
    mul $t2, $t1, $t1      #t2 = r^2
    mul $t2, $t2, $t1      #t2 = r^3
    mul $s2, $t0, $t2      #s2 = ar^3

    mul $t3, $t2, $t1      #t3 = r^4
    addi $t4, $t3, -1      #t4 = r^4 - 1
    addi $t5, $t1, -1      #t5 = r - 1
    mul $t6, $t0, $t4      #t6 = a(r^4 - 1)
    div $t6, $t5           #t6 = a(r^4 - 1)/(r - 1)
    mflo $s3               #s3 = a(r^4 - 1)/(r - 1)

    li $v0, 10
    syscall
```

Int Regs [10]

```

PC      = 4194480
EPC     = 0
Cause   = 0
BadVAddr = 0
Status  = 805371664

HI      = 0
LO      = 60

R0 [r0] = 0
R1 [at] = 268500992
R2 [v0] = 10
R3 [v1] = 0
R4 [a0] = 1
R5 [a1] = 2147480496
R6 [a2] = 2147480504
R7 [a3] = 0
R8 [t0] = 4
R9 [t1] = 2
R10 [t2] = 8
R11 [t3] = 16
R12 [t4] = 15
R13 [t5] = 1
R14 [t6] = 60
R15 [t7] = 52
R16 [s0] = 71
R17 [s1] = 156
R18 [s2] = 32
R19 [s3] = 60
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 312
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147480492
R30 [s8] = 0
R31 [ra] = 4194328

```

s_0 = eighth term (in A.P.)

s_1 = sum of the first six (in A.P.)

s_3 = fourth term (in G.P.)

s_4 = sum of first four numbers (in G.P.)

Q2 - Assembly program to implement a half adder and subtractor (using the list of the instructions given in tables 2,3 and 4).

```
#Sambit Sahoo 22277

.data

x: .byte 0
y: .byte 1

.text
.globl main
main:
    #Half Adder
    lb $t0, x           #t0 = x
    lb $t1, y           #t1 = y
    nor $t2, $t0, $0     #t2 = ~x
    nor $t3, $t1, $0     #t3 = ~y
    and $t4, $t0, $t3     #t4 = ~yx
    and $t5, $t1, $t2     #t5 = ~xy
    or $s0, $t4, $t5     #s0 = ~yx+~xy (sum)

    and $s1, $t0, $t1     #s1 = x&y (carry)

    #Half Subtractor
    move $s2, $s0         #s2 = s0 (diff)

    and $s3, $t2, $t1     #s3 = ~x&y (borrow)

    li $v0, 10
    syscall
```

Int Regs [10]

PC	= 4194392
EPC	= 0
Cause	= 0
BadVAddr	= 0
Status	= 805371664
HI	= 0
LO	= 0

```

R0  [r0] = 0
R1  [at] = 268500992
R2  [v0] = 10
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 2147480496
R6  [a2] = 2147480504
R7  [a3] = 0
R8  [t0] = 0
R9  [t1] = 1
R10 [t2] = -1
R11 [t3] = -2
R12 [t4] = 0
R13 [t5] = 1
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 1
R17 [s1] = 0
R18 [s2] = 1
R19 [s3] = 1
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147480492
R30 [s8] = 0
R31 [ra] = 4194328

```

$s_0 = \text{sum (Half Adder)}$

$s_1 = \text{carry (Half Adder)}$

$s_3 = \text{diff (Half Subtractor)}$

$s_4 = \text{borrow (Half Subtractor)}$

The example taken is $x = 0$ and $y = 1$

Q3- Assembly program to swap two register values using logical instructions given in tables 2 and 4. The values are loaded into the register using the instructions given in table 3.

```
#Sambit Sahoo 22277

.data

x: .word 10
y: .word 14

.text
.globl main
main:
    #Swap
    lw $t0, x           #t0 = x
    lw $t1, y           #t1 = y
    or $t2, $t0, $0     #t2 = x
    or $t0, $t1, $0     #t0 = y
    or $t1, $t2, $0     #t1 = x

    li $v0, 10
    syscall
```

Int Regs [10]

PC	= 4194372
EPC	= 0
Cause	= 0
BadVAddr	= 0
Status	= 805371664
HI	= 0
LO	= 0

```

R0  [r0] = 0
R1  [at] = 268500992
R2  [v0] = 10
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 2147480496
R6  [a2] = 2147480504
R7  [a3] = 0
R8  [t0] = 14
R9  [t1] = 10
R10 [t2] = 10
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147480492
R30 [s8] = 0
R31 [ra] = 4194328

```

$t_0 = 10$ (initial value)

$t_1 = 14$ (initial value)

Then we can see in the register file that its swapped.

Q4. An assembly program to compute a weighted average of four 32-bit numbers (A, B, C, and D) stored in the registers \$t0, \$t1, \$t2, and \$t3, and return the value into \$a0.

Expression: $(0.125A + 0.25B + 0.5C + 0.5D)$

Note (for Problem-4): Do not use multiplication/division instructions or floating-point registers.

```
#Sambit Sahoo 22277
```

```
.data
```

```
A: .word 100
```

```
B: .word 150
```

```
C: .word 250
```

```
D: .word 750
```

```
.text
```

```
.globl main
```

```
main:
```

```
    #Weighted Avg
```

```
    lw $t0, A           #t0 = A
```

```
    lw $t1, B           #t1 = B
```

```
    lw $t2, C           #t2 = C
```

```
    lw $t3, D           #t3 = D
```

```
    srl $t4, $t0, 3      #t4 = A/8
```

```
    srl $t5, $t1, 2      #t5 = B/4
```

```
    srl $t6, $t2, 1      #t6 = C/2
```

```
    srl $t7, $t3, 1      #t7 = D/2
```

```
    add $t8, $t4, $t5     #t8 = A/8 + B/4
```

```
    add $t9, $t8, $t6     #t9 = A/8 + B/4 + C/2
```

```
    add $a0, $t9, $t7     #a0 = A/8 + B/4 + C/2 + D/2
```

```
    li $v0, 17
```

```
    syscall
```

Int Regs [10]

PC	= 4194404
EPC	= 0
Cause	= 0
BadVAddr	= 0
Status	= 805371664
HI	= 0
LO	= 0

```

R0  [r0] = 0
R1  [at] = 268500992
R2  [v0] = 17
R3  [v1] = 0
R4  [a0] = 549
R5  [a1] = 2147480496
R6  [a2] = 2147480504
R7  [a3] = 0
R8  [t0] = 100
R9  [t1] = 150
R10 [t2] = 250
R11 [t3] = 750
R12 [t4] = 12
R13 [t5] = 37
R14 [t6] = 125
R15 [t7] = 375
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 49
R25 [t9] = 174
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 268468224
R29 [sp] = 2147480492
R30 [s8] = 0
R31 [ra] = 4194328

```

$t_0 = A$ (e.g. 100)

$t_1 = B$ (e.g. 150)

$t_3 = C$ (e.g. 250)

$t_4 = D$ (e.g. 750)

$$a_0 = (0.125A + 0.25B + 0.5C + 0.5D)(Ans - 549)$$