Laiba Fatima Khan (lk04067)

CA Lab 03 Report

Task 1a

```
addi x19, x0,1 #f
addi x20, x0, 10 #g
addi x21, x0, 2 #h
addi x22,x0, 4 #i
addi x23,x0,4 #j
#assuming that variables f to j are in registers x19 to x23
bne x22 , x23 , Else
add x19 , x20 , x21
beq x0 , x0 , Exit
Else: sub x19 , x20 , x21
Exit: # the code after if else goes here


addi x22, x0, 0 #i
addi x24, x0, 1 #k
addi x25, x0, 0xAB #address of Save
addi x1, x0, 204 #random
sw x1, 0(x25)


#assuming i and k in x22 and x24 , and the base address of Save in x25
Loop:
slli x10 , x22 , 3  #Temp reg x10 = i*8
add x10 , x10 , x25 # x10 = address of save [ i ]
lw x9 , 0( x10 ) #Temp reg x9 = save [ i ]
bne x9 , x24 , Exit #go to Exit if save [ i ] != k
addi x22 , x22 , 1 #i = i + 1
```

```
beq x0 , x0 , Loop #go to Loop
Exit:
```

Task 1b

```
#store values for x
addi x1, x0, 1 #1
addi x2, x0, 2
addi x3,x0, 3
addi x4, x0, 4

add x8, x0, x3 #x (change to check all cases)

#store a,b,c
addi x21, x0, 0 #a
addi x22, x0, 4 #b
addi x23,x0, 5 #c

beq x8, x1, L1
beq x8, x2, L2
beq x8, x3, L3
beq x8, x4, L4
add x21, x0,x0 #default a=0

L1: add x21, x22, x23  #a=b+c
beq x0, x0, Exit
L2: sub x21, x22, x23  #a=b-c
beq x0, x0, Exit
L3: add x21, x22, x22  #a=b*2
beq x0, x0, Exit
```

```
L4: div x21, x22, x2  #a=b/2
beq x0, x0, Exit
Exit:
```

Task 1c

```
#initialize i in x22
add x22, x0, x0 #i=0

#initialize sum in x23
add x23, x0, x0

#store base memory address of array "a" (4-byte integers array)
addi x1, x0, 0x200

addi x5, x0, 10 ##stores 10 needed to stop loop

Loop:
slli x10, x22, 2 # saves i*4 since 4bytes
add x10, x10, x1 #adress of a[i]
sw x22, 0(x10) #a[i]=i
add x21, x0, x22 #temp storage of a[i] value in register x2
beq x22, x5, EXIT #if i=10 loop exit
addi x22, x22, 1 #i=i+1
blt x22, x5,Loop

Loop2:
slli x10, x22, 2 # saves i*4 since 4bytes
```

```
add x10, x10, x1 #adress of a[i]
lw x9, 0(x10) #load a[i] in x9
add x23, x23, x9 #sum=sum+a[i]
beq x22, x5, EXIT #if i=10 loop exit
addi x22, x22, 1 #i=i+1
blt x22, x5,Loop2

EXIT:
add x22,x0,x0
```

Task 1d

```
#initialize i in x22
add x22, x0, x0

#initialize j in x23
add x23, x0, x0

#initialize temp in x5
add x5,x0,x0

#store base memory address of array "a" (4-byte integers array)
addi x1, x0, 0x200

addi x6, x0, 10 ##stores 10 needed to stop loop

Loop:
slli x10, x22, 2 # saves i*4 since 4bytes
add x10, x10, x1 #adress of a[i]
sw x22, 0(x10) #a[i]=i
```

```
add x21, x0, x22 #temp storage of a[i] value in register x2
beq x22, x6, EXIT #if i=10 loop exit
addi x22, x22, 1 #i=i+1
blt x22, x6,Loop


EXIT:
add x22,x0,x0 #i=0

Loop2outer:
slli x10, x22, 2 # saves i*4 since 4bytes
add x10, x10, x1 #adress of a[i]
lw x9, 0(x10) #load a[i] in x9

add x23,x0, x22 #j=i
beq x0,x0,Loop2inner



Loop2inner:
slli x11, x23, 2 # saves j*4 since 4bytes
add x11, x11, x1 #adress of a[j]
lw x8, 0(x11) #load a[j] in x8
lw x9,0(x10)
blt x9, x8, swapij

back:
addi x23, x23, 1 #j=j+1
blt x23, x6,Loop2inner
```

```
beq x22, x6, EXIT2 #if i=10 loop exit
addi x22, x22, 1 #i=i+1
blt x22, x6,Loop2outer
swapij:
add x5, x0, x9
sw x8, 0(x10)
sw x5, 0(x11)
beq x0,x0,back

EXIT2:
```