**Lab codes**

**Lab 6**

.data

M1:.word 0x1,0x2,0x3,0x4,0x5,0x6,0x7,0x8,0x9

M2:.word 0x9,0x8,0x7,0x6,0x5,0x4,0x3,0x2,0x1

M3:.word 0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0

.text

la x10,M1

la x11,M2

la x12,M3

li x13,0

li x14,9

loop:

lw x15,0(x10)

lw x16,0(x11)

add x17,x15,x16

sw x17,0(x12)

addi x13,x13,1

addi x10,x10,4 #next byte in a

addi x11,x11,4 #next byte in b

addi x12,x12,4 #next byte in c

bne x13,x14, loop

exit:nop

.data

mat1: .word 1,2,3,4,5,6,7,8,9

mat2: .word 10,11,12,13,14,15,16,17,18

.equ row1,3

.equ row2,3

.equ col2,3

res: .zero 36

.text

li x4,row1

la x1,mat1

la x3,res

nextrow: li x5,col2

la x2,mat2

nextcol: li x6,row2

mv x7,x0

dotprod: lw x8,0(x1)

lw x9,0(x2)

mul x10,x8,x9

add x7,x7,x10

addi x1,x1,4

addi x2,x2,col2\*4

addi x6,x6,-1

bne x6,x0,dotprod

sw x7, 0(x3)

addi x3,x3,4

addi x5,x5,-1

beq x5,x0,skip

addi x1,x1, -row2\*4

addi x2,x2,4-row2\*col2\*4

j nextcol

skip: addi x4,x4,-1

bne x4,x0,nextrow

.data

A:.word 27

#checking with 5

.text

la x3,A

lw x5,0(x3)

addi a0,x0,1

loop: blt x5,a0,exit

addi x5,x5,-5

beq x0,x0,loop

exit: beq x5,x0,exit1

addi a0,x0,0

exit1:nop

#if rem =0, x10=1

#if rem=1, x10=0

**LAB 9**

**9A**

.data

str:.string "madam"

.text

la x3,str

li sp,0x500

rpt:

lbu x4,0(x3) #hold first letter in string

beq x4,x0,retr #if obtained letter in string is 0

addi x8,x8,1 #counter

sb x4,-1(sp) #store in 0x500

addi x3,x3,1 #next byte

addi sp,sp,-1 #increase stack pointer

j rpt

retr:

la x3,str #reload the string

next:

lbu x6,0(sp) #sp's data

lbu x4,0(x3) #letter in string

bne x4,x6,exit #if letter is 0, exit

addi sp,sp,1 #decrease stack pointer

addi x3,x3,1 #next element in string

addi x8,x8,-1 #decrease counter

bne x8,x0,next #if counter !=0, loop back to next:

addi x1,x0,1 #else it is a palindrome

loop: j last

exit:

addi x1,x0,0 #if exit, then not a palindrome, so store 0 into x1 register

last: nop

**9B**

.data

num:.byte 0xa1

.text

la x1,num

lbu x2,0(x1)

mv x3,x2 #copy x2 into x3

srli x4,x3,1 #d1

srli x5,x3,3 #d3

srli x6,x3,4 #d4

srli x7,x3,6 #d6

xor x3,x3,x4

xor x3,x3,x5

xor x3,x3,x6

xor x6,x3,x7

andi x10,x3,1

mv x3,x2

srli x4,x3,2 #d2

srli x5,x3,3 #d3

srli x6,x3,5 #d5

srli x7,x3,6 #d6

xor x3,x3,x4

xor x3,x3,x5

xor x3,x3,x6

xor x3,x3,x7

andi x3,x3,1

slli x3,x3,1

or x10,x10,x3 #c1 c0

mv x3,x2

srli x4,x3,1

srli x5,x3,2

srli x6,x3,3

srli x7,x3,7

xor x3,x5,x4

xor x3,x3,x6

xor x3,x3,x7

andi x3,x3,1

slli x3,x3,3

or x10,x10,x3 #c2-c1 c0

mv x3,x2

srli x4,x3,4

srli x5,x3,5

srli x6,x3,6

srli x7,x3,7

xor x3,x5,x4

xor x3,x3,x6

xor x3,x3,x7

andi x3,x3,1

slli x3,x3,7

or x10,x10,x3

andi x4,x2,0x1

slli x4,x4,2

or x10,x10,x4 #c3----c2d0c1 c0

andi x4,x2,0x0e

slli x4,x4,3

or x10,x10,x4 #c3 d3 d2 d1c2 d0c1

andi x4,x2,0xf0

slli x4,x4,4

or x10,x10,x4

sw x10,4(x1)

**10 A code**

.data

array:.word 0x50,3

.text

la x8,array

lw sp,0(x8)

lw x10,4(x8)

factorial: addi sp,sp,-8

sw x1,4(sp)

sw x10,0(sp)

addi x5,x10,-1

bge x5,x0,L1

addi x10,x0,1

addi sp,sp,8

jalr(x1)

L1:

addi x10,x10,-1

jal x1,factorial

addi x6,x10,0

lw x10,0(sp)

lw x1,4(sp)

addi sp,sp,8

mul x10,x10,x6

jalr(x1)

**10B**

.data

result:.word 1

n:.word 5

.text

lw x5,n

lw x10,result

fact: beq x5,x0,exit

mul x10,x10,x5

addi x5,x5,-1

j fact

exit: sw x10,8(x3)

nop

**11A code**

.data

r1:.word 0b10101010

.text

li s0,LED\_MATRIX\_0\_BASE

li s1,LED\_MATRIX\_0\_WIDTH

li s2,LED\_MATRIX\_0\_HEIGHT

li s3,0x0a3799

#li s3,0xFFFFFF

#li s4,0x00FFFF #cyan colour

li s4, 0xffc72c

addi s5,s0,8\*4

li s6,255

li a1,0x000000

mv a0,s0

res: sw a1,0(a0)

addi a0,a0,4

bne a0,s5,res

li t1,8

mv a0,s0

start: li t1,8

on: sw s4,0(a0)

addi a0,a0,4

addi t1,t1,-1

bne t1,x0,on

li t3,8

off: sw a1,0(a0)

addi a0,a0,-4

addi t3,t3,-1

bne t3,x0,off

j start

nop

**12 CODE**

.text

li x2,3

li x3,4

li x5,5

#1

add x1,x2,x3 #7

sub x4,x1,x5 #2

#2

lw x1,0(x2)

sub x4,x1,x5

#3

sub x2,x1,x3

and x12,x2,x5

or x13,x6,x2

add x14,x12,x2

sw x14,100(x13)

#4

lw x2,20(x1)

and x4,x2,x5

or x8,x2,x6

and x9,x4,x2

sub x1,x6,x7

#1 no hazard

#add x1,x2,x3 #7

#nop

#nop

#sub x4,x1,x5 #2

#2 no hazard

#lw x1,0(x2)

#nop

#nop

#sub x4,x1,x5

#3 no hazard

#sub x2,x1,x3

#nop

#nop

#and x12,x2,x5

#or x13,x6,x2

#nop

#add x14,x12,x2

#nop

#nop

#sw x14,100(x13)

#4 no hazard

lw x2,20(x1)

nop

nop

and x4,x2,x5

or x8,x2,x6

nop

and x9,x4,x2

sub x1,x6,x7

.data

num:.byte 0xa1

.text

la x1,num

lbu x2,0(x1)

mv x3,x2

#hamming code format:

#d7d6d5d4 c3 d3d2d1 c2 d0 c1c0

#parity bits

#c0 = d0^d1^d3^d4^d6

srli x4,x3,1

srli x5,x3,3

srli x6,x3,4

srli x7,x3,6

xor x3,x3,x4

xor x3,x3,x5

xor x3,x3,x6

xor x3,x3,x7

andi x10,x3,1

#c1 = d0^d2^d3^d5^d6

mv x3,x2

srli x4,x3,2

srli x5,x3,3

srli x6,x3,5

srli x7,x3,6

xor x3,x3,x4

xor x3,x3,x5

xor x3,x3,x6

xor x3,x3,x7

andi x3,x3,1

slli x3,x3,1

or x10,x10,x3

#c2 = d1^d2^d3^d7

mv x3,x2

srli x4,x3,1

srli x5,x3,2

srli x6,x3,3

srli x7,x3,7

xor x3,x4,x5

xor x3,x3,x6

xor x3,x3,x7

andi x3,x3,1

slli x3,x3,3

or x10,x10,x3

#c3 = d4^d5^d6^d7

mv x3,x2

srli x4,x3,4

srli x5,x3,5

srli x6,x3,6

srli x7,x3,7

xor x3,x4,x5

xor x3,x3,x6

xor x3,x3,x7

andi x3,x3,1

slli x3,x3,7

or x10,x10,x3

#data bits

andi x4,x2,0x1

slli x4,x4,2

or x10,x10,x4

andi x4,x2,0x0e

slli x4,x4,3

or x10,x10,x4

andi x4,x2,0xf0

slli x4,x4,4

or x10,x10,x4

sw x10,4(x1)