**2.3**

sub x27, x28, x29

slli x27, x27, 3

add x3, x10, x27 // address of A[i-j]

ld x27, 0(x3)

sd x27, 64(x11)// B[8]= A[i-j]

**2.7**

slli x28, x28, 3

slli x29, x29, 3

add x3, x10, x28 //address of A[i]

add x4, x10, x29 //address of A[j]

ld x28, 0(x3)

ld x29, 0(x4)

add x29, x28, x29 //A[i]+A[j]

sd x29, 64(x11) // B[8]= A[i]+A[j]

**2.10**

(1)0x5000000000000000

(2)overflow

(3)0xB000000000000000

(4)not overflow

(5)0xD000000000000000

(6)overflow

**2.14**

sub x6, x7, x5 belong to R-type

**2.21**

x5>=0, go to ELSE

x6= x0 or 2 = 2

**2.29**

//assume that register x10 is n

fib:

beg x10, x0, ELSE // n==0, go to ELSE

addi x11, x0, 1

beg x10, x11, ELSE // n==1, go to ELSE

addi x2, x2, -16 //adjust stack to make room for 2 items

sd x1, 0(x2)

sd x10, 8(x2)

addi x10, x10, -1 //x10=n-1

jal x1, fib //get fib(n-1)

ld x11, 8(x2)

sd x10, 8(x2)

addi x10, x11, -2 // x10=n-2

jal x1, fib //get fib(n-2)

ld x11, 8(x2)

add x10, x10, x11 //fib(n-1) + fib(n-2)

ld x1, 0(x2)

addi x2, x2, 16 //adjust stack to delete 2 items

jalr x0, 0(x1)

ELSE:

jalr x0, 0(x1)

**2.40**

(1)2\*70%+6\*10%+3\*20%=2.6

(2)2.6\*(1-25%)=1.95

1.95-6\*10%-3\*20%=0.75

0.75/0.7=1.07

need to improve to 1.07cycles

(3)2.6\*(1-50%)=1.3

1.3-6\*10%-3\*20%=0.1

0.1/0.7=0.14

need to improve to 0.14cycles