Stamatios Morellas
COM S 321 Problem Set 2
Due: 10/1/19

Stamati Morellas

Com S 321 - Problem Set 2

2.1, 2.2, 2.6, 2.10, 2.24, 2.26, 2.32, 2.36, 2.37, 2.41, 2.42

10/1/19

2.1

$f = g + (h-5)$

$f \rightarrow x0$

$g \rightarrow x1$

$h \rightarrow x2$

Create a third register (x3) to evaluate h-5

```
SUBI x3, x2, #5
ADD x0, x1, x3
```

2.2

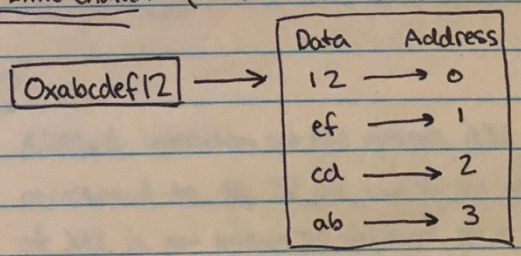
ADD f, g, h $\longrightarrow$ $f = g + h$

ADD f, i, f $\longrightarrow$ $f = i + f$

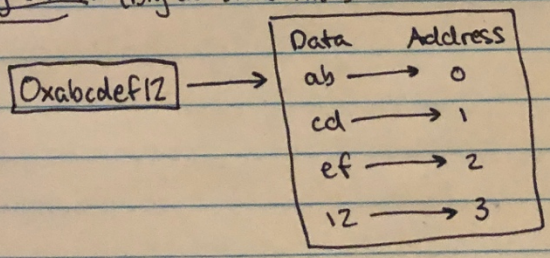
Evaluates to: $f = i + g + h$

2.6

For Little-endian: (Little-end stored first)



For Big-endian: (Big end stored first)



2.10

Values

(1) ADDI X9, X6, #8 ⟶ X9 = X6 + 8 [TypeI]

(2) ADD X10, X6, XZR ⟶ X10 = X6 + XZR ⟶ X10 = X6 [TypeR]

(3) STUR X10, [X9, #0] ⟶ Memory[X9 + 0] = X10 [Type-R]

(4) LDUR X9, [X9, #0] ⟶ X9 = Memory[X9 + 0] [Type-D]

(5) ADD X0, X9, X10 ⟶ X0 = X9 + X10 [TypeD]

For (1): Opcode: $(580)_{10}$
Source Register: X8
Target Register: X9
Immediate Field: constant 8

Opcode: Op
Source: Rn
Target: Rd/Rt
SecondSrc: Rm

For (2): Opcode: $(1112)_{10}$
Source Register: X6
Target Register: X10

For (4): Opcode: $(1986)_{10}$
Source: X9
Target: X9

For (3): Opcode: $(1984)_{10}$
Source: X9
Target: X10

For (5): Opcode: $(1112)_{10}$
Source: X9
Target: X0
second: X10

2.24.1

ARMv8 instruction set has options A32, A64, or T32, which correspond to 32, 64, and 16-bit properties. The value of X12 is not known, so therefore it is better to use A64 instruction set to accommodate for memory.

Assume
a=x12
b=0

2.24.2

```
LOOP: ADD b, b, a //b=b+X12
SUBI a, b, 1 //X12 = X12-1
B loop
Exit
```

2.26

```
for (i=0; i<a; i++)
    for (j=0; j<b; j++)
        D[4*j] = i+j
```

a → X0
b → X1
i → X10
j → X11
D[] → X2

Translation:

```
ADDI X10, 0, -1 //i=-1
[redacted]
(0) LOOP: ADDI X10, X10, 1 //i++
B.LT X2, X10, X0 //i<a
B.EQ X2, 0, exit //D[] is empty
AND X11, X11, 0 //j=0
(I) [redacted] LOOP:
        B.LT X2, X11, X1 //j<b
        B.EQ X2, 0, (0)LOOP
        ADD X2, X10, X11 //i+j
        LSL x4, X11, 4 //temp = 4·j
        ADD X3, X5, X4
        STUR x2, 0 (x3)
        ADDI X11, X11, 1 //j++
        jump to ILOOP
```

2.32

```
int f(int a, int b, int c, int d) {
    return g(g(a,b), c+d)
}
```

a →
c → x9
d → x10
c+d → 4(SP)

LEGv8:

```
f:
    ADDI SP, SP, -8
    STUR x17, 0 (SP)  //return address
    ADD [illegible] 4(SP), x9, x10  //c+d
    STUR x18, x9
    BR func g
    MOVE x19, X11
    LDUR X20, 4 (SP)
    BR func g
    [illegible]
    LDUR X17, 0(SP)
    ADDI SP, SP, 8
    B X17
```

2.36

2.36.1

Big-Endian Machine: 11223344

2.36.2

Little-Endian Machine: 88776655

2.41.1

$$\text{Initial CPI} = \left(1 \times \frac{500}{900}\right) + \left(3 \times \frac{100}{900}\right) + \left(10 \times \frac{300}{900}\right)$$

$$= \frac{30}{9} + \frac{1}{3} + \frac{5}{9} = \frac{38}{9} = 4.22$$

New Instructions: $0.75 \times 500 = 375 \Rightarrow 125\%$ 25%

$$\text{Initial Execution} = \frac{38}{9} \times \text{freq.}_{\text{clk}} \times 900 = 3800\,\text{f}$$

$$\text{New CPI} = \left(1 \times \frac{375}{775}\right) + \left(3 \times \frac{100}{775}\right) + \left(10 \times \frac{300}{775}\right)$$

$$= \frac{375}{775} + \frac{300}{775} + \frac{3000}{775} = \frac{3675}{775} = 4.74$$

New Execution: $\frac{3675}{775} \times \text{freq}_{\text{clk}} \times 775 \times 1.10 = 4042.5\text{f}$

Execution Time goes up so it is not a good design choice.

2.41.2 New CPI = 0.5 arith instructions

[redacted] New CPI $= \frac{0.5 \times 500}{900} + \frac{3 \times 100}{900} + \frac{10 \times 300}{900}$

$= \frac{250}{900} + \frac{300}{900} + \frac{3000}{900} = \frac{3550}{900} = \frac{355}{90} = 3.944$

Overall Speedup: $\frac{4.22}{3.944} = 1.0699$

10X : CPI = ~~0.01 × 500 / 900 + 3 × 100 / 900 + 10 × 300 / 900~~

~~= 0.5 / 900 + 300 / 900 + 3000 / 900 = 3305 / 900~~

$\frac{50}{900} + \frac{300}{900} + \frac{3000}{900} = \frac{3350}{900} =$

Speedup = ~~(CPI×1 − CPI×10) / CPI×10~~

= ~~(3550 − 3350) / 3350~~

$38 / 33.5 = 1.134$

2.42.1

[redacted]

Average CPI = (0.7 × 2 cycles) + (0.1 × 6 cycles) + (0.2 × 3 cycles)

= **2.6 cycles**

[redacted]

2.42.2

25% improvement → (New) Avg CPI = 1.95

$0.7x + 0.1 \times 6 + 0.2 \times 3 = 1.95$

$0.7x + 0.6 + 0.6 = 1.95$

$0.7x = 0.75$

$x = 1.07$ → number of cycles ≈ 1 (avg)

2.42.3

50% improvement → New AVG CPI = 1.3

$0.7x + 0.6 + 0.6 = 1.3$

$0.7x = 0.1$

$x = 1 + 0.14285$

$= 1.14285$ cycles

Extra Credit: