91. 0b00101011 → $00101011_2 = 43_{10}$

0xA5 → $1010\ 0101_2 = -91_{10}$

0b10101101 → $10101101_2 = -83_{10}$

0x6B → $0110\ 1011_2 = 107_{10}$

0xFA → $1111\ 1010_2 = -6_{10}$

0x80 → $1000\ 0000 = -128_{10}$

92. 00101011_2 <=> 0x005B

$1010\ 0101_2$ <=> 0xFFA5

10101101_2 <=> 0xFFAD

0x6B <=> 0x006B

0xFA <=> 0xFFFA

0x80 <=> 0xFF80

93. operandos unsigned

O overflow só é detetado quando o bit carry é '1'; esta situação ocorre quando o resultado obtido não cabe na gama de valores que estamos a considerar.

94. operandos signed

O overflow é detetado quando o carry-in do MSB é diferente do carry-out: $\underbrace{C_{n-1}}_{\text{carry-in}} \oplus \underbrace{C_n}_{\text{carry-out}} = 1$! OVERFLOW

95. i. \$s0 = 0x70000000 \$s1 = 0x0FFFFFFF

ii. \$s0 = 0x40000000 \$s1 = 0x40000000

a) add \$t0, \$s0, \$s1

i)
```
  0x70000000 → 1879048192₁₀
+ 0x0FFFFFFF → 268435455
  0x7FFFFFFF
```

ii)
```
  0x40000000
+ 0x40000000
  0x80000000
```

b) Em i) o resultado era esperado (a soma de 2 nº's positivos resultou num nº positivo); em ii) a soma de 2 positivos resultou num negativo, logo ocorreu overflow.

c) sub $t0, $s0, $s1

i) 0x70000000
- 0x0FFFFFFF
0x160000001

ii) 0x40000000
- 0x40000000
0x00000000

d) Para i) obteve-se o resultado esperado

Para ii) obteve-se o resultado esperado

e) i) add $t0, $s0, $s1 → $t0=0x7FFFFFFF
add $t0, $t0, $s1

$t0= 0x8FFFFFFE

C.A
0x7FFFFFFF
+ 0x0FFFFFFF
0x8FFFFFFE

ii) $t0 = 0x80000000

C.A
0x80000000
+ 0x40000000
0xC0000000

$t0 = 0xC0000000

f) Para i) ocorreu overflow pois a soma de 2 nºs positivos resultou num negativo.
Para ii) o resultado era o esperado: somou-se 1 número negativo com 1 positivo em que o 1º tem maior valor absoluto; o resultado é [redacted] 1 valor negativo

96. 1º operando → n bits
2º " → m bits

Ex: 11 → 2 bits
x 111 → 3 bits
10101 → 5 bits

R: (n+m) bits que o resultado necessita

1111 → 4 bits
x 111 → 3 bits
1101001 → 7 bits

97. Decomposição em nativas

a) mul \$5, \$6, \$7

mult \$6, \$7

mflo \$5

b) la \$t0, label (c/ label = 0x00400058)

lui \$1, 0x0040

ori \$t0, \$1, 0x0058

c) div \$2, \$1, \$2

div \$1, \$2

mflo \$2

d) rem \$5, \$6, \$7

div \$6, \$7

mfhi \$5

e) ble \$8, 0x16, target

ori \$1, \$0, 0x16

slt \$2, \$1, \$8

beq \$2, \$0, target

f) bgt \$4, 0x3F, target

ori \$1, \$0, 0x3F

slt \$2, \$1, \$4

bne \$2, \$0, target

98. mul \$5, \$6, \$7

$\$6 = \underbrace{0xFFFFFFFE}_{-2}$ $\$7 = \underbrace{0x00000005}_{5}$

$-2 \times 5 = -10 \Rightarrow \$5 = 0xFFFFFFF6$

99. div \$5, \$6, \$7

$\$6 = \underbrace{0xFFFFFFF0}_{-16}$ $\$7 = \underbrace{0x00000003}_{3}$

$-16/3 = -5$

divisão inteira

$\$5 = 0xFFFFFFFB$

rem \$5, \$6, \$7

$-16 \% 3 = -1$ $\$5 = 0xFFFFFFFF$

$(-5) \times 3 - 1 = -16$

100. $\$t0 = \$t2 / \$t3$ $\$t1 = \$t2 \% \$t3$

div \$t2, \$t3

mflo \$t0 # \$t0 = \$t2/\$t3

mfhi \$t1 # \$t1 = \$t2 % \$t3

101. 1º Divide-se o dividendo pelo divisor em módulo

2º O quociente tem sinal negativo se o sinal do dividendo e do divisor forem diferentes; se estes forem iguais, o sinal do quociente é positivo.

3º O resto tem o mesmo sinal que o dividendo

102. $t0 = -7 $t1 = 2   div $t0, $t1

HI: -1   Lo: -3

103. $t0 = 0xFFFF FFF9 = -7

$t1 = 0x0000 0002 = 2   HI: -1   Lo: -3

104. $5 = -9 $10 = 2   rem $6, $5, $10

$6 = -1

105. Multiplicador → 32 bits

Multiplicando → 32 bits

ALU → 64 bits

106. a) $a0 = 0x7FFFFFF1 $a1 = 0x0000000E

signed

$v0 = 0

$1 = 0x7FFFFFFF

slt: $1 = 0

addu: $1 = 0x7FFF FFFF

xor: $1 = 0x0000000E

slt: $1 = 0

jr $ra

$v0 = 0

Não ocorreu overflow

unsigned

$v0 = 0

nor: $1 = 0xFFFFFFF1

sltu: $1 = 0

jr $ra

$v0 = 0

Não ocorreu overflow

106.

b) $a0 = 0x7FFF FFF1 $a1 = 0x0000 000F

Signed

$v0 = 0

xor: $1 = 0x7FFFFFFE

slt: $1 = 0

addu: $1 = 0x80000000

xor: $1 = 0xFFFFFFF1

slt: $1 = 1

ori: $v0 = 1

Ocorreu overflow

Unsigned

$v0 = 0

nor: $1 = 0xFFFFFFF0

sltu: $1 = 0

jr $ra

$v0 = 0

Não ocorreu overflow

c) $a0 = 0xFFFF FFF1 $a1 = 0xFFFF FFFF

Signed

$v0 = 0

xor: $1 = 0x0000000E

slt: $1 = 0

addu: $1 = 0x1 FFFFFFF0

xor: $1 = 0x1 00000001

slt: $1 = 0

jr $ra

$v0 = 0

Não ocorreu overflow

Unsigned

$v0 = 0

nor: $1 = 0x00000000

sltu: $1 = 1

ori: $v0 = 1

Ocorreu overflow

d) $a0 = 0x80000000 $a1 = 0x80000000

Signed

$v0 = 0

xor: $1 = 0x00000000

slt: $1 = 0

addu: $1 = 0x1 00000000

xor: $1 = 0x1 80000000 < 0

slt: $1 = 1

ori: $v0 = 1

Ocorreu overflow

Unsigned

$v0 = 0

nor: $1 = 0x7FFFFFFF

sltu: $1 = 1

ori: $v0 = 1

Ocorreu overflow

107. (Semelhante ao 106.)

108. Estas sub-rotinas não terminais / sub-rotinas folha, não necessitam de salvaguardar qualquer registo, apenas a função que "chama" estas sub-rotinas tem de salvaguardar registos.

109. Formato IEEE 754, precisão simples

No máximo 6 casas decimais $\rightarrow \left\lfloor 23 \cdot \frac{\log_{10}(2)}{\log_{10}(10)} \right\rfloor = 6$

110. Formato IEEE 754, precisão dupla

No máximo 15 casas decimais $\rightarrow \left\lfloor 52 \cdot \frac{\log_{10}(2)}{\log_{10}(10)} \right\rfloor = 15$

111. $19{,}1875_{10}$ no formato IEEE

Precisão simples:

Parte inteira: 10011_2

Parte fracionária: 0011_2

$10011.0011 = 1.00110011 \times 2^4$

$E = 127 + 4 = 131_{10} = 10000011_2$

C.A

$$\begin{array}{r} 0{,}1875 \\ \times \quad 2 \\ \hline 0{,}3750 \end{array} \qquad \begin{array}{r} 0{,}3750 \\ \times \quad 2 \\ \hline 0{,}7500 \end{array} \qquad \begin{array}{r} 0{,}7500 \\ \times \quad 2 \\ \hline 1{,}5000 \end{array} \qquad \begin{array}{r} 0{,}5000 \\ \times \quad 2 \\ \hline 1{,}0000 \end{array}$$

S	E	f	
0	10000011	00110011000000000000000	$\boxed{0x41558000}$

Precisão dupla:

$E = 1023 + 4 = 1027_{10} = 10000000011$

S	E	f	
0	10000000011	$\underbrace{00110011000\ldots000}_{52\ bits}$	$\boxed{0x4033300000000000}$

112. Determinar em decimal as seguintes quantidades formato IEEE 754, precisão simples

a) 0xC19AB000 = 1100 0001 1001 1010 1011 0000 0000 0000

S=1 E=10000011 = 131 f=0011010 1011

Exp = 131 − 127 = 4

$$1.00110101011 \times 2^4 = 10011.0101011 \times 2^0$$

Parte inteira: $10011_2 = 19_{10}$

" fracionária: $0101011_2 = 0.335938$

−19,335938

b) 0x80580000 = 1000 0000 0101 1000 0000 0000 0000 0000

S=1 E=00000000 f=1011000 0000 0000 0000 0000

Exp = E − 127
= −127 → expoente verdadeiro = −126

Representação desnormalizada!!!

$$0.1011 \times 2^{-126} \xRightarrow{\text{com sinal } (-)} -8{,}081524 \times 10^{-39}$$

113. \$f0 = 0x416A0000 \$f2 = 0xC0C00000

a) abs.s \$f4, \$f2

\$f2 = 1100 0000 1100 0000 0000 0000 0000 0000

\$f4 = 0100 0000 1100 0000 0000 0000 0000 0000
= 0x40C00000

b) neg.s \$f6, \$f0

\$f0 = 0100 0001 0110 1010 0000 0000 0000 0000

\$f6 = 1100 0001 0110 1010 0000 0000 0000 0000
= 0xC16A0000

c) sub.s \$f8, \$f0, \$f2

$\$f0 = 0100\ 0001\ 0110\ 1010\ 0000\ 0000\ 0000\ 0000$

$S=0$

$E = 10000010_2 = 130_{10} \Rightarrow Exp = 3$

$f = 0.1101010_2$

$$1.1101010 \times 2^3$$

$\$f2 = 1100\ 0000\ 1100\ 0000\ 0000\ 0000\ 0000\ 0000$

$S=1$

$E = 10000001_2 = 129_{10} \Rightarrow Exp = 2$

$f = 0.1_2$

$$1.1 \times 2^2$$

$$1.1101010 \times 2^3 - 1.1 \times 2^2 = (1.1101010 - \underbrace{0.11}_{<0}) \times 2^3$$

$$= (1.1101010 + 0.11) \times 2^3$$

$$= 10.1001010 \times 2^3 = 1.01001010 \times 2^4$$

C.A

$$\begin{array}{r} {\scriptstyle 1\ 1} \\ 1.1101010 \\ +\ 0.11 \\ \hline 10.1001010 \end{array}$$

$|\$f0| > |\$f2| \Rightarrow$ O sinal do resultado é positivo

$$\underset{S}{0}\quad \underset{E}{10000011}\quad \underset{f}{01001010\,000000000000000}$$

$Exp = 4 \rightarrow E = 10000011$

$\$f8 = 0x41A50000$

d) sub.s \$f10, \$f2, \$f0

\$f0 = 1.1101010×2^3 $\quad$ \$f2 = 1.1×2^2

>0 $\quad$ <0

$$1.1 \times 2^2 - 1.1101010 \times 2^3 = (\underbrace{0.11}_{<0} - \underbrace{1.1101010}_{>0}) \times 2^3 = -(0.11 + 1.1101010) \times 2^3$$

$= -\ 10.1001010 \times 2^3$

$$\begin{array}{r} 0.1100000 \\ +\,1.1101010 \\ \hline 10.1001010 \end{array}$$

$= -\ 1.0100 1010 \times 2^4$

1 $\quad$ 10000011 $\quad$ 01001010000000000000000

S $\quad$ E $\quad$ f

\$f10 = 0xC1A50000

e) add.s \$f12, \$f0, \$f2

$$1.1101010 \times 2^3 + \underbrace{1.1}_{<0} \times 2^2 = (1.1101010 - 0.11) \times 2^3 =$$

$= 1.0001010 \times 2^3$ $\quad$ Exp = 3 ⇒ E = 10000010

|\$f0| > |\$f2| ⇒ o sinal do resultado é positivo

C.A

$$\begin{array}{r} 1.1101010 \\ -\,0.1100000 \\ \hline 1.0001010 \end{array}$$

0 $\quad$ 10000010 $\quad$ 00010100000000000000000

S $\quad$ E $\quad$ f

\$f12 = 0x410A0000

f) mul.s \$f14, \$f0, \$f2 $\quad (1.1101010 \times 1.1) \times 2^{3+2} =$

$= 10.1011111 \times 2^5 = 1.01011111 \times 2^6$ $\quad$ Exp = 6 ⇒ E = 10000101

C.A

$$\begin{array}{r} 1.1101010 \\ \times\,1.1000000 \\ \hline 0\,000\,0000 \\ (.\,.\,.\,.\,.\,.\,.\,.) \\ 111010\,1.0 \\ +\,1110101\,0.0 \\ \hline 10.1011111\,0 \end{array}$$

→ tudo '0'

o sinal do resultado é negativo pois o multiplicando e o multiplicador têm sinais opostos

1 $\quad$ 10000101 $\quad$ 01011111000000000000000

S $\quad$ E $\quad$ f

\$f14 = 0xC2AF8000

g) div.s \$f16, \$f0, \$f2

$(1.1101010 / 1.1) \times 2^{3-2} = 1.00111 \times 2$

$Exp = 1 \Rightarrow E = 10000000$

O resultado tem sinal negativo

1	10000000	00111000000000000000000
S	E	f

\$f16 = 0xC01C0000

C.A

1.1101010 | 11.1
1.001110
-1 10
0 010
-00
101
-011
0100
-0011
11
-11
000
-00
0

h) div.s \$f18, \$f2, \$f0

$(1.1 / 1.1101010) \times 2^{2-3} =$

C.A

i) cvt.d.s \$f20, \$f2

\$f2 = 1100 0000 1100 0000 0000 0000 0000 0000

1	10000001	100 0000 0000 0000 0000 0000
S	E	f

$E = 129 \Rightarrow Exp = 2$

Para precisão dupla: $E = Exp + 1023 = 2 + 1023 = 1025_{10} \Rightarrow 10000000001_2$

1	10000000001	$\overbrace{10000\ldots0}^{52 \text{ bits}}$
S	E	f

= 0xC018000000000000

\$f21 = 0xC0180000 \$f20 = 0x00000000

j) cvt.w.s \$f22, \$f0

$\$f0 = \underbrace{0}_{S}\,\underbrace{100\ 0001\ 0}_{E}\,\underbrace{110\ 1010\ 0000\ 0000\ 0000\ 0000}_{f}$

$E = 10000010_2 = 130_{10} \Rightarrow Exp = 3$

$\underbrace{1.1101010 \times 2^3}_{>0} = 14,625$

$trunc.(14,625) = 14_{10}$
$= 1110_2$

\$f22 = 0x0000000E