COMP2421 Computer Organization
Assignment 1

Wang Ruijie 22103808D

22 February 2024

1 Question with Short Answers

1. $\texttt{FD2}_{(16)} = \texttt{1111 1101 0010}_{(2)} = -2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^4 + 2^1_{(10)} = -46_{(10)}$.

2. The number is negative since the sign bit is 1. The biased exponent is $\texttt{1000 1001}_{(2)} = 137_{10}$, while the bias equals today $2^{8-1} - 1 = 127$. Thus, the real exponent is $137 - 127 = 10$. The fraction is $\texttt{111 1100 1101 0000 0000 0000}_{(2)}$. Therefore, the number is $-1 \times \texttt{1.111 1100 1101 0000 0000 0000} \times 2^{\texttt{10}}{}_{(2)} = -\texttt{1111 1100 110.1}_{(2)} = -2022.5_{(10)}$.

3. In the first expression, $\texttt{0111}_{(2)} = 7$, $\texttt{1111}_{(2)} = -1_{(10)}$, and the result of the first expression is $\texttt{0110}_{(2)}$ with a carry forward bit 1. If we ignore, it will be $\texttt{0110}_{(2)} = 6_{(10)}$, which does not lead to overflow since the 4-bit result equals to the sum of the operands $(7 + (-1) = 6)$.

   In the second expression, $\texttt{1110}_{(2)} = -2_{(10)}$, and $\texttt{1000}_{(2)} = -8_{(2)}$. The result of the second expression is also $\texttt{0110}_{(2)} = 6_{(10)}$ with a carry forward bit 1. However, since the result is not equal to the sum of the operands $(-2 + (-8) = -10 \neq 6)$, it leads to overflow.

4. ```
   sltu $t3, $t2, $t1
   ```

5. ```
   addi $sp, $sp, -8
   sw $s1, 4($sp)
   sw $s2, 0($sp)
   ```

2 Translate Pseudo-instructions

(1) ```
   srl $t2, $t1, 7
   sll $t3, $t1, 25
   or $t1, $t2, $t3
   ```

(2) ```
   sll $t1, $t2, 5
   sub $t1, $t1, $t2
   ```

(3)
```
lui $t2, 0x0001
ori $t2, $t2, 0x0002
add $t3, $t1, $t2
lw $t4, 0($t3)
```

3 Translate MIPS Program into C Program

Condition 1: `(x >= y || z <= w) && x == z`

Condition 2: `z != w`

Code 1: z = y - 2;

Code 2: x = y + z;

4 Understand MIPS Code

(1) The statement implements a while loop. The loop will continue to execute if `$t1 <= $v1` and `$t0 != $ai`.

(2) `$t0` stores the addresses of the elements, sequentially from the second to the last in the array, while `$t1` stores the value of the element whose address is stored by $$t0$$. In other words, $$t1$$ stores the values ofthe elements, sequentially from the second to the last in the array.

`$v0` stores the address of the currently found maximum, while `$v1` stores the value of the currently found maximum of the array.

(3) The two instructions are used to update the address of the currently found maximum and its value, storing them in the two registers.

(4) The usage of the instruction is to examine if the last element of the array has been reached and compared with the currently found maximum. If not, the loop will continue to execute to repeat the previous process.

(5) The function is designed for iterating through the array, comparing each of the elements with the currently found maximum to find and return the maximum value and its address.

(6) The content of `$v0` will be `0x20060018`, and the content of `$v1` will be 106.