Assignment 2 - Handwritten Part (50%)

Question 1

Following is a C code of comparing value in comp with k.

```c
int i = 0;
while(comp[i] == k){
    i += 1;
}
```

And here is the assembly code version of the code above.
(Array comp's base address are saved in x25.)

```
      add    x22, x0,    x0    // i = 0
Loop: slli   x10, x22,   2     // x10 = i * 4
      add    x10, x10,   x25   // x10 = address of comp[i]
      lw     x9,  0(x10)       // x9 = comp[i]
      bne    x9,  x24,   Exit  // go to Exit if comp[i] != k
      addi   x22, x22,   1     // i = i + 1
      beq    x0,  x0,    Loop  // go to Loop
Exit:
    // code..
```

There are two labels: **Loop & Exit** inside this code.

Try to convert this assembly code into **one label** version.
(with annotation is better)

Question 2

Following are three group of hexadecimal, please

1. convert them into binary
2. translate them into assembly code
3. explain what does these instruction actually do.

```
0x00A484B3
0x40A48533
0x40A484B3
```

Question 3

Bob wants to use RISC-V assembly to calculate f = x + min(y, z). Suppose the input x, y, z are first placed in register x10, x11, x12, and the return value f should be placed in x10 at the end of the program. Here is his code:

```
Main: jal  x1,  Min
      add  x10, x10,  x12
      jal  x0,  Exit
```

```
Min:  bge  x10, x11,  L
      jalr x0,  0(x1)

L:    add  x10, x11,  x0
      jalr x0,  0(x1)
```

```
Exit:
```

However, the output of this program is incorrect. Please help him fix his code to produce the correct output. You can only **insert a few lines** in the main function. Modify the exsiting code is **not allowed!!**