Problem FC-7 (5 parts)

Part A Write a C code fragment equivalent to the following MIPS code. Register $2 holds the variable “N” and $4 holds the variable “S”.

```c
int Foo(N) {
    int S = 0; // # of ones
    do {
        S += N & 1;
        N >>= 1;
    } while (N)
    return (S);
}
```

Part B Briefly describe what the code fragment in Part A computes.

This routine counts the number of ones in the binary representation of N.

Part C Write a MIPS code fragment that jumps to instruction address 0x8A6B9C2C.

```c
lui $1, 0x8A6B # load 8A6B into upper 16 bits of $1
ori $1, $1, 0x9C2C # place 9C2C in the lower 16 bits
jr $1 # jump to address in $1
```

Part D Write a two instruction MIPS code fragment that branches to label Target if register $4 is less than or equal to $5.

```c
slt $3, $5, $4 # is $5 < $4?
beq $3, $0, Target # if not, branch to Target
```

Part E Turn this compound predicate if-then-else statement into the equivalent nested if-then-else statement which does not use compound predicates (i.e., do not use the && and || operators).

```c
if ((a==3) && (b>0) || (c<10))
    z = 10;
else
    z = 5;
```

Equivalent C code:

```c
if (a==3)
    if (b>0)
        z = 10;
    else if (c<10)
        z = 10;
    else
        z = 5;
else if (c<10)
    z = 10;
else
    z = 5;
```