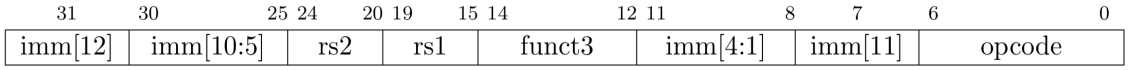

Figure 1: SB-Type format. Used for branch instructions in RISC-V.

```
0x80004000          add  t0, a0, x0
0x80004004          add  a0, x0, x0
0x80004008          bgt  a1, x0, left
0x8000400C          slli t4, a0, 2
                    ...
0x80004040          add  t1, x0, x0
0x80004044          slli t2, t1, 2
0x80004048          add  t3, t0, t2
0x8000404C          lw   t4, 0(t3)
                    ...
0x80004060          add  a0, a0, t4
0x80004064          addi t1, t1, 1
0x80004068          blt  t1, a1, right
0x8000406C          jr   ra
```

Figure 2: RISC-V code with missing labels.

Instruction Address	Instruction Binary	Immediate value	Target Address
0x80004008	0x04B04263	0x00000044	0x8000404C
0x80004068	0xFCB34EE3	0xFFFFFFDC	0x80004044

Question 1 (12 points): Figure 1 shows the SB-Type format that is used for branch instructions in RISC-V. Figure 2 shows RISC-V assembly code in which the labels for the target of the two branch instructions have not been written. The execution of every branch instruction requires the creation of a 32-bit immediate value that is used to determine the address of the instruction that is the target of the branch instruction. This question asks you to determine both the 32-bit immediate value and the address of the target instruction for each of the two branch instructions in the code of Figure 2. Express all four values in hexadecimal notation and write them in the Table above.

Branch at 0x80004008:
Instruction Binary: 0x04B04263
Instruction Binary: 0000 0100 1011 0000 0100 0010 0110 0011
Instruction Binary: 0 000010 01011 00000 100 0010 0 1100011
Immediate Fields in Instruction: 0 0 000010 0010
Immediate Value: 0x00000044
Target address: 0x8000404C

Branch at 0x80004068:
Instruction Binary: 0xFCB34EE3
Instruction Binary: 1111 1100 1011 0011 0100 1110 1110 0011

Instruction Binary: `1 111110 01011 00110 100 1110 1 1100011`
Immediate Fields in Instruction: `1 1 111110 1110`
Immediate Value: `0xFFFFFFDC`
Target address: `0x80004044`