

```

100011 00000 00001 0000000000000000 // lw $1, 0($0); $1 = Mem[0] = 255;
100011 00000 00010 0000000000000001 // lw $2, 1($0); $2 = Mem[1] = 256;
100011 00000 00011 0000000000000111 // lw $3, 7($0); $3 = Mem[7] = 3;
100011 00011 00100 0000000000001011 // lw $4, 11($3); $4 = Mem[14] = 261;
000000 00001 00010 00110 00000 100000 // add $6, $1, $2; $6 = 255 + 256 = 511
000000 00100 00001 00111 00000 100010 // sub $7, $4, $1; $7 = 261 - 255 = 6
100011 00111 10000 0000000000000101 // lw $16, 5($7); $16 = Mem[11] = 7
000000 00001 00100 00101 00000 100100 // and $5, $1, $4; $5 = 255 & 261 = 5
000000 10000 00010 01000 00000 100101 // or $8, $16, $2; $8 = 7 | 256 = 263
101011 00101 01000 0000000000000111 // sw $8, 7($5); Mem[12] = 263
001000 00001 01010 0000000100000000 // addi $10, $1, 256; $10 = 255 + 256 = 511
001100 00100 01011 0000000011111111 // andi $11, $4, 255; $11 = 261 & 255 = 5
001101 10000 01100 0000000100000000 // ori $12, $16, 256; $12 = 7 | 256 = 263
100011 00000 01001 0000000000001100 // lw $9, 12($0); $9 = Mem[12] = 263;
000100 01001 01000 0000000000000001 // beq $9, $8, 1; goto PC + 8;
000000 00000 01011 01101 00100 000000 // sll $13, $11, 4; $13 = $11 << 4
000000 00000 01011 01110 00100 000000 // sll $14, $11, 4; $14 = $11 << 4 = 80
000000 00000 01010 01111 00100 000010 // srl $15, $10, 4; $15 = $10 >> 4 = 31
000000 01111 01110 10001 00000 101010 // slt $17, $15, $14; $17 = 1
000000 01111 10000 10010 00000 101010 // slt $18, $15, $16; $18 = 0
000010 0000000000000000000000010110 // j 22
000000 01111 01110 10010 00000 101010 // slt $18, $15, $14; $18 = 1 (line 21)
000011 0000000000000000000000011000 // jal 24 (goto line 24)
000100 10000 01011 0000000000000011 // beq $16, $11, 3; goto PC + 16;
001000 01011 01011 0000000000000010 // addi $11, $11, 2; $11 = $11 + 2 = 7
000000 11111 00000 00000 00000 001000 // jr $31 (goto line 23)
001000 01011 01011 0000000000000010 // addi $11, $11, 2; $11 = $11 + 2 = 9
001000 01011 01011 0000000000000010 // addi $11, $11, 2; $11 = $11 + 2 = 9

```

Total test: lw sw add sub and or slt beq sll srl j jal jr andi ori addi