Question 5 (15 points): Translate the following C code into MIPS. Assume that `str` is a null-terminated character array, declared as `char str[]`, and that the address of the base of `str` is in `$s0`. Do not use pseudoinstructions in your code.

```
int i = 0;
while (str[i] != 0) {
    if (str[i] > 90) {
        str[i] = str[i] - 32;
    }
    i++;
}
```

Since the array contains chars, we need to load/store and index bytes, not words.

```
        addi $s1, $zero, $zero     # int i = 0
loop:   add  $t0, $s0, $s1         # get address of str[i]
        lbu  $t1, 0($t0)           # get value of str[i]
        beq  $t1, $zero, end       # if 0, end
        slti $t2, $t1, 91          # check for < 91
        bne  $t2, $zero, end_loop  # branch if < 91
        addi $t1, $t1, -32         # str[i] = str[i] - 32
        sb   $t1, 0($t0)           # store updated str[i]
end_loop: addi $s1, $s1, 1         # i++
        j loop
end:    # next instruction is here
```

Q: what does this do?