

# Str\_ucase Procedure

```
397 ;-----
398 ; Str_ucase Procedure
399 ; Converts a null-terminated string to uppercase.
400 ; Returns: nothing
401 ;-----
402 Str_ucase PROC USES eax esi,
403     pString:PTR BYTE    ; Pointer to the string
404
405     mov esi, pString    ; Initialize esi with the address of the string.
406 L1:
407     mov al, [esi]        ; Load the character from the string.
408     cmp al, 0            ; Check if it's the end of the string.
409     je L3                ; If yes, exit the loop.
410     cmp al, 'a'          ; Compare the character with 'a'.
411     jb L2                ; If it's below 'a', go to L2.
412     cmp al, 'z'          ; Compare the character with 'z'.
413     ja L2                ; If it's above 'z', go to L2.
414
415     and BYTE PTR [esi], 11011111b
416     ; Convert the character to uppercase by clearing the 6th bit.
417 L2:
418     inc esi              ; Move to the next character.
419     jmp L1               ; Repeat the loop.
420 L3:
421     ret                  ; Return when the entire string is converted to uppercase.
422 Str_ucase ENDP
```

The first part of the code sets up the procedure. It expects a pointer to a string (null-terminated) in pString, and it initializes esi with the address of this string.

Here's what's happening within the loop:

`mov al, [esi]` loads the character at the memory address pointed to by esi into the al register.

`cmp al, 0` checks if the character is the null terminator (end of the string). If it is, the loop exits (`je L3`).

`cmp al, 'a'` compares the character with the ASCII value of 'a'. If the character is below 'a', it jumps to L2.

`cmp al, 'z'` compares the character with the ASCII value of 'z'. If the character is above 'z', it also jumps to L2.

**and BYTE PTR [esi], 11011111b** converts the character to uppercase by clearing the 6th bit (bit 5) in the ASCII code. This changes lowercase letters to uppercase.

**L2:** marks the location to which the code jumps when the character is not a lowercase letter, ensuring it's skipped.

**inc esi** increments the esi register to point to the next character in the string.

**jmp L1** jumps back to the beginning of the loop, continuing the process until the end of the string is reached.

**Finally, L3:** is the label that is reached when the end of the string is detected. At this point, the loop exits, and the procedure returns.

The entire string is converted to uppercase by iterating through each character, and no value is returned; the original string is modified in memory.