```asm
global writeline
extern _write

section .text
; rdi = address of null-terminated string
; Writes a null-terminated string to stdout
writeline:              ; copy string pointer to rsi
    mov rsi, rdi        ; rsi = pointer to string (needed for _write later)
    xor rcx, rcx        ; rcx = counter for string length, start at 0

.calc_len:
    cmp byte [rsi], 0 ; check if current byte is null terminator
    je .send           ; if null terminator, jump to send
    inc rsi            ; move to next character
    inc rcx            ; increment length counter
    jmp .calc_len      ; repeat loop

.send:
    mov rdx, rcx       ; rdx = length of string (number of bytes to write)
    mov rsi, rdi       ; rsi = pointer to string (restore original pointer for _write)
    mov rdi, 1         ; rdi = file descriptor 1 (stdout)
    call _write        ; call system _write to write string to stdout
    ret                ; return to caller
```