

# Introduzione7 Librerie

venerdì 10 ottobre 2025 15:53

Andiamo ora a relazionare il software applicativo, le librerie standard e le librerie di sistema. Quindi per **libreria standard** (standard del linguaggio) : definisce i servizi standard per la programmazione in una data tecnologia (specifico linguaggio di programmazione). Quindi si intendono i famosi moduli già programmati. Ogni funzione della libreria standard potrebbe usare le system call. Vediamo in dettaglio :



Vediamo un esempio :

```
#include <stdio.h>

int main(int x, char** y){
    char c;

    while(1){
        scanf("%c",&c);
        printf("%c",c);
    }
}
```

```
#include <unistd.h>

int main(int x, char** y){
    char c;

    while(1){
        read(0,&c,1);
        write(1,&c,1);
    }
}
```

```
//neverending echo from the standard input to the standard output
//please compile with "gcc -c" "and "ld" or with "-nostartfiles"

.file "asm-terminal-echo.S"
.text
.globl _start
.type _start, @function

_start:
    sub $0x4, %rsp //eserve 4 bytes for hosting chars to read/write
.loop:
    mov $0x0, %rax //from this line we pack registers for calling the read syscall
    mov $0x0, %rdi
    mov %rsp, %rsi
    mov $0x1, %rdx
    syscall
    mov $0x1, %rax //from this line we update registers for calling the write syscall
    mov $0x1, %rdi
    syscall
    jmp .loop
```

Il primo programma rappresenta un classico programma C che può essere compilato da entrambi i sistemi operativi. Nel secondo esempio vado ad usare direttamente le funzioni del SO. Nel dettaglio il secondo snippet viene scritto **esclusivamente** per SO unix. Mentre nell'ultimo si ha la funzione assembler