



POLITECNICO
MILANO 1863

Programmazione modulare: creazione di librerie

Fondamenti di Informatica A.A. 2017/18

Nel corso dei laboratori avete più volte dovuto importare librerie:

- stdio.h
- stdlib.h
- math.h

Queste librerie sono sempre disponibili.

È possibile creare una vostra libreria contenente funzioni che potrete riutilizzare agevolmente.

Promemoria:

Le librerie statiche sono linkate dal linker e incluse nel codice eseguibile.

Se 5 programmi importano la stessa libreria, ciascuno ne avrà una copia nel suo eseguibile.

Esempio – Libreria util.h

Abbiamo una serie di funzioni utili che vorremmo raccogliere in
una libreria **util.h** in modo da utilizzarle in **mio_prog.c**

Esempio – Interfaccia util.h

Step 1

Creare un file **util.h** che costituirà l'interfaccia della libreria, con le dichiarazioni delle funzioni e dei tipi necessari.

Inserite sempre `#ifndef ... #endif` per evitare che il preprocessore inserisca la vostra libreria più volte.

Esempio – Interfaccia util.h

```
#ifndef _UTIL_H_
#define _UTIL_H_

    // definisco una costante
#define MAX_FOO 20

    // definisco un tipo
struct foo_struct { int x; float y; };
typedef struct foo_struct foo_struct;

    // definisco una funzione
int foo(float y, float z);

#endif
```

Esempio – Implementazione util.c

Step 2

Creare un file **util.c** che includa **util.h** e contenga l'implementazione delle funzioni che la libreria esporta.

Esempio – Implementazione util.c

```
// importo l'interfaccia
#include "util.h"

// questa variabile non compare in util.h quindi
// non la esporto ma la posso usare
// nell'implementazione
int total_foo;

// implemento altre funzioni che però non esporto
int useful_stuff(float y, float z) { ...
}

// implemento la funzione esportata
int foo (float y, float z) { ...
}
```

Esempio – Object file util.o

Step 3

Compilare **util.c** in un object file **util.o**

```
gcc -o util.o -c util.c
```

Il file **util.o** potrà ora essere linkato in altri programmi

Esempio – Usiamo la libreria

Step 4

Importiamo la libreria in **mio_prog.c** e usiamo quanto esporta:
funzioni, tipi e costanti

Esempio – Usiamo la libreria

```
// importo l'interfaccia
#include "util.h"

int main() {

    // Uso la costante
    for (int i = 0; i < MAX_FOO; i++) {

        // Uso la funzione
        result = foo(a,v) ;
        ...
    }
    ...
}
```

Esempio – Link della libreria

Step 5

Linkare programma e libreria:

```
gcc mio_prog.c util.o
```

Il comando salverà l'usuale file eseguibile **a.out**

Riassumendo

- | | |
|------------------------------|-------------------------|
| 1. Definire l'interfaccia | util.h |
| 2. Implementare la libreria | util.c |
| 3. Creare l'object file | gcc -o util.o -c util.c |
| 4. Implementare un programma | mio_prog.c |
| 5. Linkare l'object file | gcc mio_prog.c util.o |