

Verifica conoscenze preliminari

COMPILATORI E LINKER

Avete usato il compilatore gcc?

MODULI IN C

Se un programma è composto da più file sorgenti in linguaggio C, allora è possibile compilare quei file separatamente e poi linkarli insieme in un secondo momento per creare un file eseguibile?

VARIABILI IN ALTRI MODULI

E' possibile, in linguaggio C, definire una variabile in un modulo ed utilizzarla in un modulo diverso?

E' possibile definire una variabile globale in un modulo ed impedire che venga usata dal codice collocato in un diverso modulo?

MACRO IN C

Sapete cosa sono le macro nel linguaggio C? Sapete scrivere una macro?

Test #1

programma formato da due moduli

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

void main(void)
{
    printf( "%d \n", funzione( "ciao gatto" ) );
}
```

```
/* file funzione.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

int funzione( char *str ) {
    if ( str==NULL) return(0);
    else return ( strlen(str) );
}
```

- 1) Se compilo **separatamente** i due moduli e poi li linko insieme, accade qualche errore o ci sono warning oppure riesco a creare un eseguibile?
- 2) Se accade qualche problema, come posso correggerlo?

Test #1 - soluzione

programma formato da due moduli

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

int funzione( char *str );

void main(void)
{
    printf( "%d \n", funzione( "ciao gatto" ) );
}
```

```
/* file funzione.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <stdlib.h>

int funzione( char *str ) {
    if ( str==NULL) return(0);
    else return ( strlen(str) );
}
```

Viene sollevato un warning che indica un potenziale problema.

Occorre inserire il prototipo della funzione prima del punto nel main in cui si chiama la funzione, altrimenti il compilatore nel main non conosce la funzione.

Test #2

programma formato da due moduli

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

int funzione (char *str);

void main(void)
{
    printf( "%d \n", funzione( "ciao gatto" ) );
}
```

```
/* file funzione.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

static int funzione( char *str ) {
    if ( str==NULL) return(0);
    else return ( strlen(str) );
}
```

- 1) Se compilo separatamente i due moduli e poi li linko insieme, accade qualche errore oppure riesco a creare un eseguibile?
- 2) Se accade qualche errore, come posso correggerlo?

Test #2 - soluzione

programma formato da due moduli

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

int funzione (char *str);

void main(void)
{
    printf( "%d \n", funzione( "ciao gatto" ) );
}
```

```
/* file funzione.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

/* togliere keyword static */

int funzione( char *str ) {
    if ( str==NULL ) return(0);
    else return ( strlen(str) );
}
```

Durante il linking accade un errore.

Occorre togliere la parola chiave static davanti alla dichiarazione della funzione poiché quella keyword impedisce di usare la funzione fuori dal modulo in cui è dichiarata.

Test #3

macro in linguaggio C

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

#include "macro.h"

void main(void)
{
    int a=3;
    funzione( a );
}
```

```
/* file macro.h */

#define funzione( X )      \
    printf ("%i \n", X);  \
    printf ("fine \n")
```

- 1) questo programma è formato da uno o da due moduli?
- 2) Se compilo e linko tutto quanto, accade qualche errore oppure riesco a creare un eseguibile?
- 3) La macro è fatta bene?

Test #3 - soluzione macro in linguaggio C

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

#include "macro.h"

void main(void)
{
    int a=3;
    funzione( a );
}
```

```
/* file macro.h */

#define funzione( X )      \
    printf ("%i \n", X);  \
    printf ("fine \n")
```

Il programma è formato da un solo modulo. Il file .h è incluso dal modulo main, non è un modulo separato.

Compilando e linkando il main non accadono errori.

Ma la macro non è fatta bene perché in alcuni casi non può essere usata, vedi esempio successivo.

Test #4

macro in linguaggio C

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

#include "macro.h"

void main(void)
{
    int a=3;
    if (a==4 ) funzione( a );
    else      funzione( -a );
}
```

```
/* file macro.h */

#define funzione( X )      \
    printf ("%i \n", X);  \
    printf ("fine \n")
```

- 1) con questa modifica nel main, se compilo il main e lo linko accade qualche errore oppure riesco a creare un eseguibile?
- 2) Se accade un errore, come devo modificare la macro per risolvere?

Test #4 - soluzione macro in linguaggio C

```
/* file main.c */
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>

#include "macro.h"

void main(void)
{
    int a=3;
    if (a==4 ) funzione( a );
    else      funzione( -a );
}
```

```
/* file macro.h */

#define funzione( X )      \
                            \
    do {                    \
        printf ("%i \n", X); \
        printf ("fine \n");  \
    } while (0)
```

Compilando il main, la macro produce nel main un errore sintattico.
Occorre modificare la macro inserendola in un ciclo while che fa solo un giro.