Sistemi Operativi

Laurea in Ingegneria Informatica
Università Roma Tre
Docente: Romolo Marotta

Richiami di programmazione C

1. Compilazione
2. Puntatori
3. printf
4. scanf
5. Layout

Dal sorgente all'eseguibile

PREPROCESSORE: tutte le direttive vengono sostituite con il contenuto del file. Ad esempio: la direttiva #include<stdio.h>s

a.out: per compilare in file c

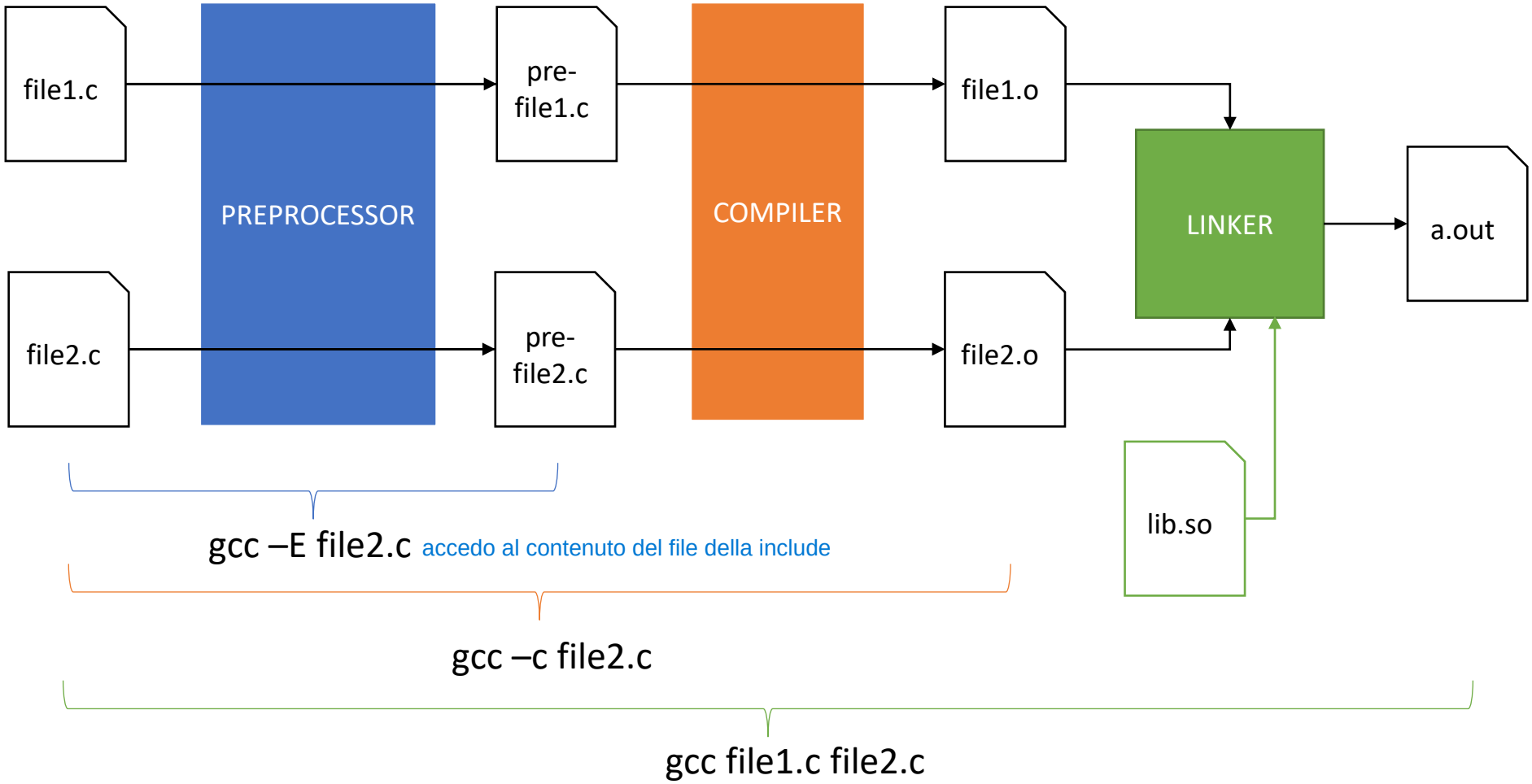
gcc -DITERATIONS=10 simple.c: con questo comando stiamo definendo una variabile

i file oggetti formano l'eseguibile

!. :esegue l'ultima riga

COMPILER: compila ogni file e genera ogni file oggetto

-o : ci dice qual'è l'output



Esempi 1&2

I puntatori

- Il linguaggio di programmazione C permette al programmatore di accedere esplicitamente allo spazio di indirizzamento di un processo
- Alcuni operatori:
  - * dato un puntatore accede al contenuto della variabile referenziata
  - & data una variabile ne ottiene il puntatore


```
printf("%d\n", var);
printf("%p\n",&var);
```

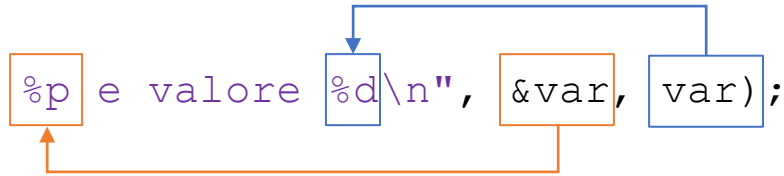
```
printf("%d\n", *var_ptr);
printf("%p\n", &var_ptr);
```

printf

... : che potrebbe prendere un numero variabile di parametri

- int printf (const char * format, ...);
- stampa su standard output la stringa *format*
- può prendere argomenti addizionali che saranno stampati secondo la formattazione specificata dalla stringa *format*
- alcuni specificatori:
  - %c : un carattere
  - %s : sequenza di caratteri terminanti con '\0' (Stringa)
  - %p : un puntatore
  - %d : signed int decimale
  - %u : unsigned int decimale
  - %x : esadecimale unsigned
  - %o : ottale unsigned

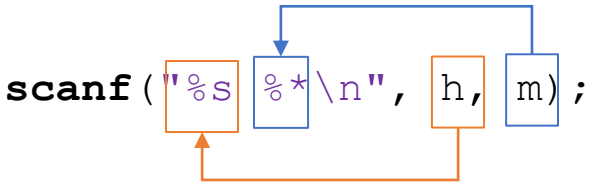
```
printf("La variabile 'var' ha indirizzo %p e valore %d\n", &var, var);
```



Esempio 3

scanf

- int scanf(const char * format, ...);
- scansiona l'input in accordo alla stringa *format*
- la stringa *format* nel può prendere argomenti addizionali che saranno stampati secondo la formattazione specificata dalla stringa *format*



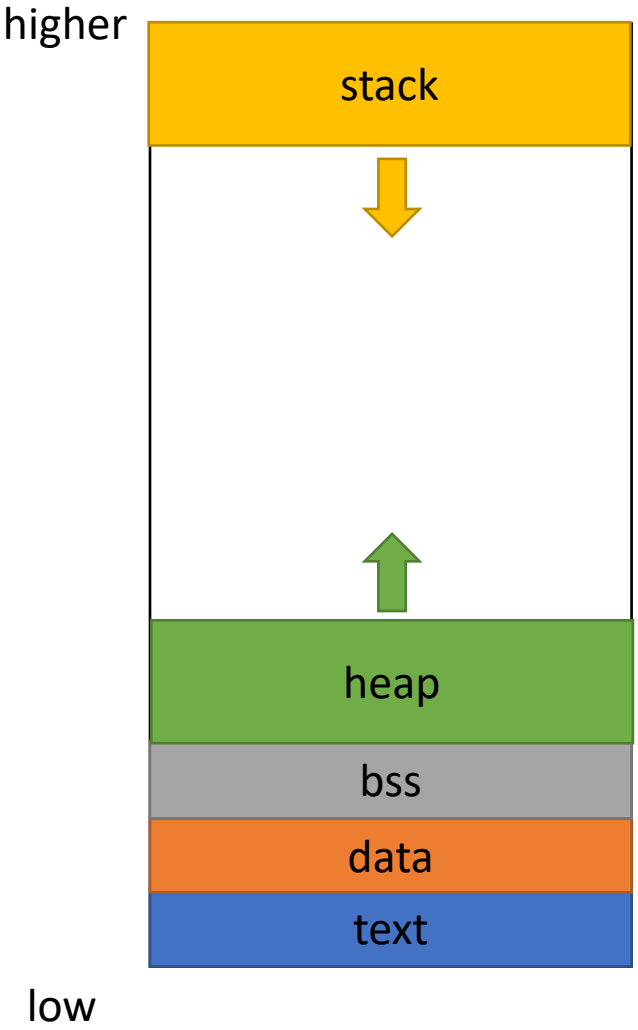
- il prototipo di uno specificatore di formato è:

```
%[*][width][length]specifier
```

%* :quel specificatore non deve essere memorizzato in nessuno variabile

Esempio 4

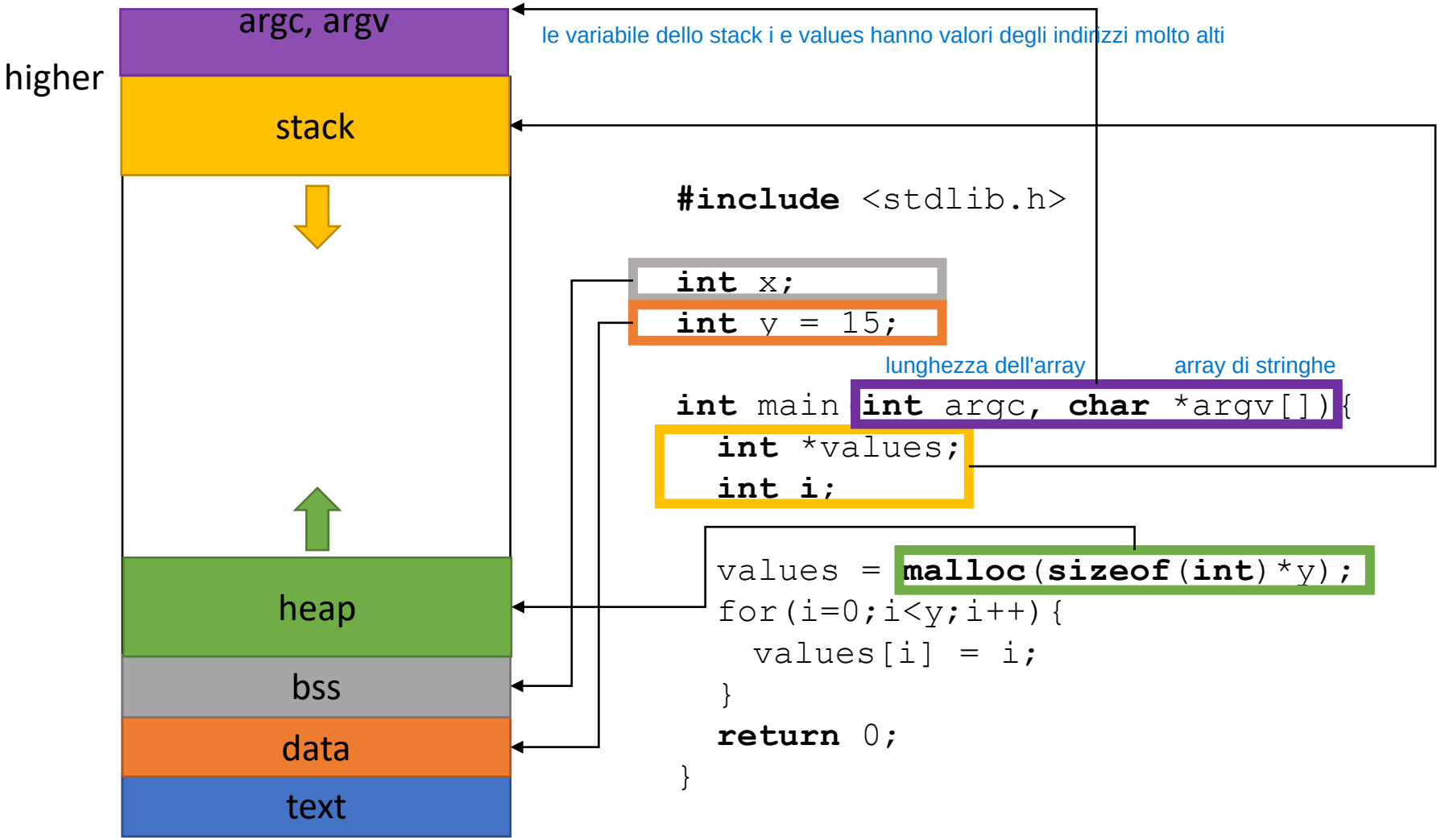
Layout di un programma C



- **Text:** istruzioni eseguibili
- **Data:** dati inizializzati
- **Block started by symbol (BSS):** dati non inizializzati o initizializzati al valore zero
- **Heap:** sezione di dati allocati dinamicamente
- **Stack:** per chiamate a procedura, passaggio parametri, indirizzo di ritorno, variabili locali

Layout di un programma C

low e higher ci l'ordine degli indirizzi in ordine crescentes



Esempio 5