Lo Stupidario di Fondamenti 2

Raccolta semiseria di errori comuni e stili di programmazione da evitare.

Daniele Paolo Scarpazza
Politecnico di Milano
Ultimo aggiornamento: 2004-03-03

Nota

- In caso di discrepanza fra Scarpazza e altre fonti di informazione, ha ragione Scarpazza.

;-)

Tipo di ritorno void per main

```
#include <stdio.h>

void main(int argc, char* argv[])

{

  printf("Ciao, mondo!");

}
```

```
program.c: In function `main':
program.c:4: warning: return type of `main' is not `int'
```

Stringa letterale non terminata

```
#include <stdio.h>

int main(int argc, char* argv[])

{

  printf("Questa stringa non è terminata);

}
```

```
program.c:5:10: missing terminating " character
program.c: In function `main':
program.c:6: error: syntax error before '}' token
```

Errore di battitura in parola chiave

```
#include <stdio.h>

int main(int argc, char * argv[])
{
  int i=0;

  whil(i>=0)
  {
    printf("Voglio stampare questa stringa dieci volte.\n");
    printf("E invece il programma non compila...");
    i--;
  }
}
```

```
program.c: In function `main':
program.c:8: error: syntax error before '{' token
program.c: At top level:
program.c:13: error: syntax error before '}' token
```

Dichiarazione di funzione con prototipo incompleto (1/2)

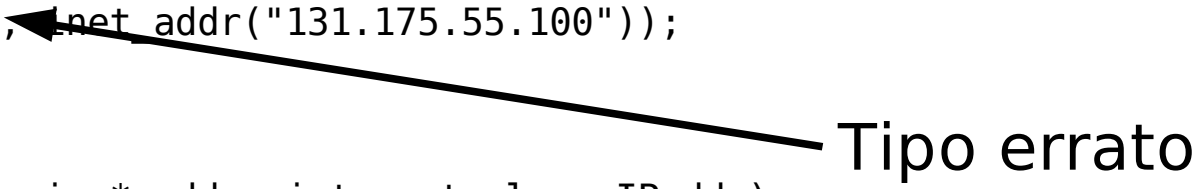
```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>

#define PORT 4000
#define MAXCONN 5
#define DIMBUF 6

void addr_initialize(); /* prototipo incompleto */

int main(int argc, char * argv[])
{
 struct sockaddr_in server_addr;
 addr_initialize(&server_addr, "80", inet_addr("131.175.55.100"));
 return 0;
}

void addr_initialize(struct sockaddr_in * addr, int port, long IPaddr)
{
 addr->sin_family = AF_INET;
 addr->sin_port = htons((u_short) port);
 addr->sin_addr.s_addr = IPaddr;
}
```



Il compilatore non può fare controllo di tipo sui parametri.
Nessun errore, nessun warning! Bizzarrie a runtime.
Morale: evitare assolutamente i prototipi incompleti.

Dichiarazione di funzione con prototipo incompleto (2/2)

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>

#define PORT 4000
#define MAXCONN 5
#define DIMBUF 6

void addr_initialize(struct sockaddr_in * addr, int port, long IPaddr);

int main(int argc, char * argv[])
{
 struct sockaddr_in server_addr;
 addr_initialize(&server_addr, "80", inet_addr("131.175.55.100"));
 return 0;
}

void addr_initialize(struct sockaddr_in * addr, int port, long IPaddr)
{
 addr->sin_family = AF_INET;
 addr->sin_port = htons((u_short) port);
 addr->sin_addr.s_addr = IPaddr;
}
```

Ho completato il prototipo

Tipo errato

```
program.c: In function `main':
program.c:14: warning: passing arg 2 of `addr_initialize' makes integer from
pointer without a cast
```

Stringa letterale senza delimitatori

```
 1  #include <stdio.h>
 2  #include <sys/socket.h>
 3  #include <netinet/in.h>
 4
 5  #define SERVER 131.175.55.100
 6  #define PORT 4000
 7
 8  void addr_initialize(struct sockaddr_in *indirizzo, int port, long IPaddr);
 9
10  int main(int argc, char* argv[])
11  {
12    int sd;
13    int error;
14    struct sockaddr_in server_addr;
15
16    addr_initialize(&server_addr, PORT, inet_addr(SERVER));
17
18    sd    = socket(PF_INET,SOCK_STREAM,0);
19    error = connect(sd,(struct sockaddr*) &server_addr, sizeof(server_addr));
20    if (error==0)
21      puts("Ho eseguito la connessione");
22    else
23      puts("Errore di connect");
24    close(sd);
25  }
26
27  void addr_initialize(struct sockaddr_in * addr, int port, long IPaddr) { ... }
```

```
program.c:16:49: too many decimal points in number
```

Punti-e-virgola indesiderati (1)

```
#include <stdio.h>

int main(int argc, char * argv[])
{
  int i=10;

  while (i>=0);
  {
    printf("Vorrei stampare questa stringa dieci volte.\n");
    i--;
  }
}
```

Dov'è l'errore?
Quali sono le conseguenze?

Punti-e-virgola indesiderati (2)

```
#include <stdio.h>

int main(int argc, char * argv[])
{
  int i;

  for (i=0; i < 10; i++);
  {
    printf("Avrei voluto stampare questa stringa dieci volte.\n");
  }
}
```

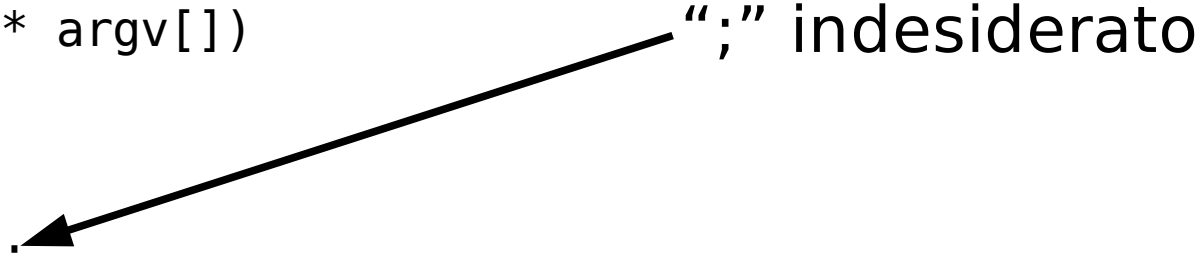
Dov'è l'errore?
Quali sono le conseguenze?

Punti-e-virgola indesiderati

```
#include <stdio.h>

int main(int argc, char * argv[])
{
  int i;

  for (i=0; i < 10; i++);
  {
    printf("Avrei voluto stampare questa stringa dieci volte.\n");
    printf("E invece viene stampata una volta sola...");
  }
}
```



Conflitto macro / parametro formale

```
 1  #include <stdio.h>
 2  #include <sys/socket.h>
 3  #include <netinet/in.h>
 4
 5  #define SERVER "131.175.55.100"
 6  #define PORT 4000
 7
 8  void addr_initialize(struct sockaddr_in *indirizzo, int PORT, long IPaddr);
 9
10  int main(int argc, char* argv[])
11  {
12    int sd;
13    int error;
14    struct sockaddr_in server_addr;
15
16    addr_initialize(&server_addr, PORT, inet_addr(SERVER));
17    sd    = socket(PF_INET, SOCK_STREAM, 0);
18    error = connect(sd, (struct sockaddr*) &server_addr, sizeof(server_addr));
19    /* ... */
20    close(sd);
21  }
22
23  void addr_initialize(struct sockaddr_in * addr, int PORT, long IPaddr)
24  {
25   addr->sin_family = AF_INET;
26   addr->sin_port = htons((u_short) PORT);
27   addr->sin_addr.s_addr = IPaddr;
28  }
```

```
program.c:8: error: syntax error before numeric constant
program.c:27: error: syntax error before numeric constant
program.c: In function `addr_initialize':
program.c:29: error: `addr' undeclared (first use in this function)
program.c:29: error: (Each undeclared identifier is reported only once
program.c:29: error: for each function it appears in.)
program.c:31: error: `IPaddr' undeclared (first use in this function)
```

Parentesi non bilanciate

```
 1     #include <stdio.h>
 2     #include <sys/socket.h>
 3     #include <netinet/in.h>
 4     #define PORT 4000
 5     #define MAXCONN 5
 6     #define DIMBUF 6
 7
 8     void addr_initialize(struct sockaddr_in *indirizzo, int port, long IPaddr);
 9
10     int main(int argc, char * argv[])
11     {
12       int sd, new_sd;
13       struct sockaddr_in server_addr, client_addr;
14       int ric, i, client_len = sizeof(client_addr);
15       char buf[DIMBUF];
16
17       addr_initialize(&server_addr, PORT, INADDR_ANY);
18       sd = socket(PF_INET,SOCK_STREAM,0);
19       bind(sd,(struct sockaddr*) &server_addr,sizeof(server_addr));
20       listen(sd,MAXCONN);
21       while (1) {
22         puts("Attendo richieste di connessione...");
23         new_sd=accept(sd,(struct sockaddr*) &client_addr, &client_len);
24         while ((ric=recv(new_sd,buf,DIMBUF,0))>0) {
25           printf("\nHo ricevuto %d caratteri: ",ric);
26           for (i=0; i<ric; i++) printf("%c", buf[i]);
27           close(new_sd);
28       }
29       return 0;
30     }
31
32     void addr_initialize(struct sockaddr_in * addr, int port, long IPaddr) { ... }
```

```
program.c: In function `main':
program.c:32: error: syntax error at end of input
```

Errore di linker per errore di battitura nel nome di funzione

```
 1    #include <stdio.h>
 2    #include <sys/socket.h>
 3    #include <netinet/in.h>
 4    #define PORT 4000
 5    #define MAXCONN 5
 6    #define DIMBUF 6
 7
 8    void addr_initialize(struct sockaddr_in *indirizzo, int port, long IPaddr);
 9
10    int main(int argc, char * argv[])
11    {
12      int sd, new_sd;
13      struct sockaddr_in server_addr, client_addr;
14      int ric, i, client_len = sizeof(client_addr);
15      char buf[DIMBUF];
16
17      addr_initialize(&server_addr, PORT, INADDR_ANY);
18      sd = socket(PF_INET,SOCK_STREAM,0);
19      bind(sd,(struct sockaddr*) &server_addr, sizeof(server_addr));
20      listen(sd,MAXCONN);
21      while (1) {
22        new_sd = accept(sd, (struct sockaddr*) &client_addr, &client_len);
23        while ( (ric = recv(new_sd, buf, DIMBUF, 0))>0 ) {
24          printf("\nHo ricevuto %d caratteri: ",ric);
25          for (i=0; i<ric; i++) printf("%c", buf[i]);
26        }
27        close(new_sd);
28      }
29      return 0;
30    }
31
32    void add_initialize(struct sockaddr_in * addr, int port, long IPaddr) { ... }
```

```
/tmp/ccecaGaq.o(.text+0x26): In function `main':
: undefined reference to `addr_initialize'
collect2: ld returned 1 exit status
```

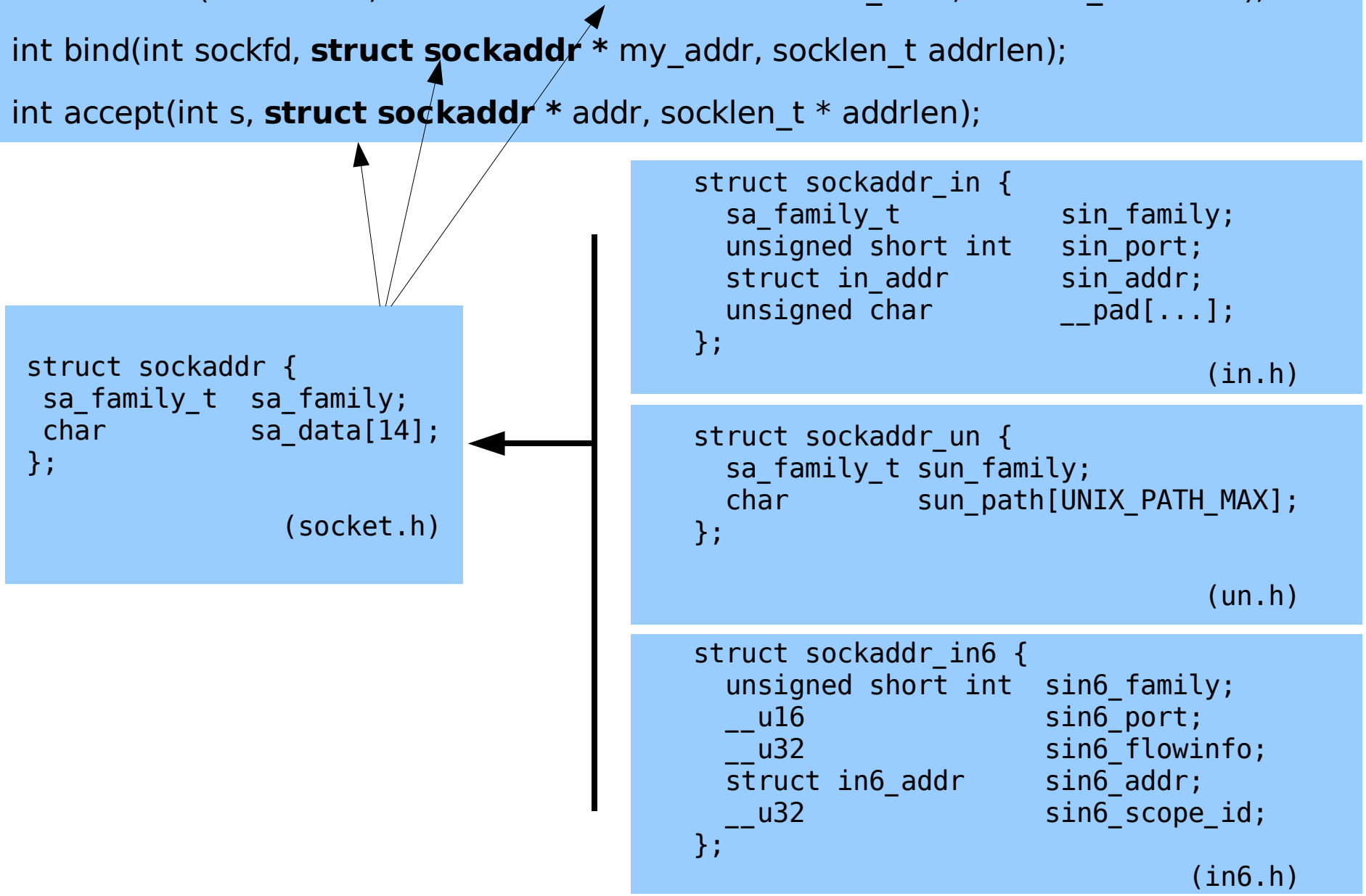
Warning per assenza di cast legittimo

```
 1     #include <stdio.h>
 2     #include <sys/socket.h>
 3     #include <netinet/in.h>
 4     #define PORT 4000
 5     #define MAXCONN 5
 6     #define DIMBUF 6
 7
 8     void addr_initialize(struct sockaddr_in *indirizzo, int port, long IPaddr);
 9
10     int main(int argc, char * argv[])
11     {
12       int sd, new_sd;
13       struct sockaddr_in server_addr, client_addr;
14       int ric, i, client_len = sizeof(client_addr);
15       char buf[DIMBUF];
16
17       addr_initialize(&server_addr, PORT, INADDR_ANY);
18       sd = socket(PF_INET, SOCK_STREAM, 0);
19       bind(sd, &server_addr, sizeof(server_addr));
20       listen(sd,MAXCONN);
21       while (1) {
22         new_sd=accept(sd, (struct sockaddr*) &client_addr, &client_len);
23         while ( (ric=recv(new_sd, buf, DIMBUF, 0))>0 ) {
24           printf("\nHo ricevuto %d caratteri: ",ric);
25           for (i=0; i<ric; i++) printf("%c", buf[i]);
26         }
27         close(new_sd);
28       }
29       return 0;
30     }
31
32     void add_initialize(struct sockaddr_in * addr, int port, long IPaddr) { ... }
```

```
program.c: In function `main':
program.c:19: warning: passing arg 2 of `bind' from incompatible pointer type
```

Il motivo del casting (struct sockaddr *)

- int connect(int sockfd, const **struct sockaddr** * serv_addr, socklen_t addrlen);
- int bind(int sockfd, **struct sockaddr** * my_addr, socklen_t addrlen);
- int accept(int s, **struct sockaddr** * addr, socklen_t * addrlen);



```
struct sockaddr {
 sa_family_t  sa_family;
 char         sa_data[14];
};
```

(socket.h)

```
struct sockaddr_in {
  sa_family_t          sin_family;
  unsigned short int   sin_port;
  struct in_addr       sin_addr;
  unsigned char        __pad[...];
};
```

(in.h)

```
struct sockaddr_un {
  sa_family_t sun_family;
  char        sun_path[UNIX_PATH_MAX];
};
```

(un.h)

```
struct sockaddr_in6 {
  unsigned short int  sin6_family;
  __u16               sin6_port;
  __u32               sin6_flowinfo;
  struct in6_addr     sin6_addr;
  __u32               sin6_scope_id;
};
```

(in6.h)