

Breve ripasso sul linguaggio C

Moreno Marzolla

Dipartimento di Informatica—Scienza e Ingegneria (DISI)

Università di Bologna

<https://www.moreno.marzolla.name/>

Copyright © 2021

Moreno Marzolla, Università di Bologna, Italy

<https://www.moreno.marzolla.name/teaching/LabASD/>



This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0) License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Ambiente di sviluppo

- È possibile usare l'ambiente di sviluppo preferito, purché il compilatore supporti lo standard **ANSI C (C90)**
 - Io userò il compilatore GCC a riga di comando in ambiente Linux
- Per chi vuole lavorare dalla riga di comando:
 - Windows: consiglio mingw <http://mingw-w64.org>
 - MacOSX: consiglio XCode che fornisce un compilatore a riga di comando
- Esempio

```
$ gcc -std=c90 -Wall -Wpedantic prova.c -o prova
```

Rispetta lo
standard ANSI C

Abilita tutti i
warning

Abilita ancora
più *warning*

Algo...ati

Ambiente di sviluppo

- I programmi dovrebbero compilare *senza warning*
 - i *warning* sono "avvertimenti" che non pregiudicano la compilazione del codice
 - però l'esperienza insegna che indicano quasi sempre un *problema logico grave* nel codice
- Non tutti i *warning* sono uguali
 - MS Visual Studio segnala che alcune funzioni (es., `fopen()`) sono "deprecate"; tali warning possono essere ignorati
 - Alcune funzioni (es., `scanf()`) restituiscono un valore che può essere ignorato, ma il compilatore potrebbe segnalare un warning che può essere ignorato

Regole di base

- Codice ordinato e indentato correttamente

```
#include <stdio.h>
int main( void ) { int i = 1; while ( i <= 10 )
{ printf("%d\n", i*i); i = i + 1; } return 0; }
```

NO 

```
#include <stdio.h>
int main( void )
{
    int i = 1;
    while ( i <= 10 ) {
        printf("%d\n", i*i);
        i = i + 1;
    }
    return 0;
}
```

SÌ 

The International Obfuscated C Code Contest <http://ioccc.org/>

```
#define/**/Alan/**/(fflush(0),j=c=0;++c<b[6]+7;v=b[c]^(b[c+
char*H="close\r\nContent-Type:text/html\r\n\r\n<canvas id=\
'c'width='128'height='128'style='width:256px;height:256px'\
><script>x=y=-1;v=window.c;v.onmousemove=function(e){x=e.p\
ageX-v.offsetLeft;y=e.                pageY-v.offsetTop};w=\
new WebSocket('ws'+                location.href.subs\
tr(4));w.binary"                "Type='arraybuff\
er';w.onmess"                "age=function(\
e){c=v.getC"                "ontext('2d')\
;b=c.crea"                "teI"                "mageData(12\
8,128);u=                new Uint8Array(d=e.                data);b.dat\
a.set(u)                ;c.putImageData(b,0,0)                ;u[0]=x/2;\
u[1]=y/2                ;w.send(d.slice(0,x<0?0:                2));x=y=-\
1}</scr"                "ipt">,b[9999];float*e,d[                65536],u,v
,lu,lv,                z;void*f;typedef unsigned                long long
1;l*p,t[                99]={0x67452301,0xEFCDAB89,                0x98BADCFE
,308438*                881,3285377520                },i,j,k
#include                /io*/                <stdio.h>                /*turing*/
void s(){                ;                for(                k                =0;k++<7 *                2                *9);for(i=
0;++i<127;                )                e=d +                i*                4                +512*k,1[e]                +=(i-u)*i
-u)+(k-v)*                (k-                v                )<20?*e+=UV_DROP:0;]*q=                m,                g=1LL<<32;
l(h(1                v,l                (a)){                return(v<<a)|                (v%g)>>(32-a                )                )                );int main
(int c/*ha                r*/,                char**y){for(e=d;fgets(b                ,m[97]=480,\
stdin)&&*b                >31;                )for(                i=j=0;n=b[i],i<18                ?                (1)"sec-webs\
ocket-key:                "[i]                ==(n|32):n-32?                m[j/4]                |(j<24?n:(1)
"258EAF5-                "[E914-47DA-95CA-C5AB0DC85B"                "11\x80"[j-24])
<<(3-j*4)*                8                ,++j<61:1;)*i++;for(j=3;--j;                )}{for(i=4;+i<74
;)*i>9?p=q+i-                2                ,p[8]=h(p[5                ]^*p^p[-6]^p[-8]
#define/*hah                aha                <--mouth*/                laplacian(u)l#u
,1):(t[i]=t[9                -i]                );for(p=t+7                ,i=-1;+i<82;k
=*p++,p[2]=h(p[1],5)                +p[-3]+n+                (i/+                20%2?*p^k^p[-
2]:(*p&k)|((i<                20?*~p:*p|k)&p                [-2                ]))*q++,t[+
81-i]+=i>76)*                (*                p=                h(*                p,30*(i<80)
))for(;n-k;n=((                1LL                +9<<i/20
)/8)<<60)/k+k)/2)                k=n                ;}for                (i=0;i<57;
i++)i<30?q[i%5*                16/                3+                i/5]                |=t[i%5]%g<<4
>>(30+i%5%3*2-i/5                *6)&                63,0                :(n                =*q++,b[i-30]=
n<62?n-37+"fl!"                [n                /26]:n                *4-                205);for(prin\
t("HTTP/1.1                "/"*                |                *"/d"                "OK\r\nConnecti\
on:%s%.27s=\r\n\r\n"                ,101                +99!*                *m,!*m:H:"upgrade\
\r\nUpgrade:webso"                "c"                "ket\r\nSec-Websocket-Accep\
t:";b);e-d<65536;                e+=                4)e[1]=(*e=UV_BACKGROUND);for
(*m;c=0){for(f=c                2                %0:fopen(y[1],"r");f&&fscanf(f,
"%f,%f,"                ,&u,&v)>0;)*s                ();for(j<SPEED*4;j+=2)for(i=0;i<
63504;e+=5-j%4*2,z=                TIMESTEP,*e=u+z*(elta_u),e[1]=v+z*
(elta_v),i+=4)for(c=                0;c+2;u=v,v=e[4],lu=lv,lv=e[-508]+*e
e[8]+e[516]-v*4)e=d+i+512+i/504*8+j%4-c--;for(;n=c%4,65544>
c;+c)putc(c>?n>2||u=d[c-n-7]*4*(n<2?n?RGB)>255?255:
#define/*x*/Turing(x)4)=getchar()&127)u=v;if(b[6]>1)s();}}
u<0?0:u:c<0?124-c*3:c==5);/* -- The word "genius" is */ for
```

Alan Turing (1912--1954)

<http://ioccc.org/2015/endoh1/prog.c>

```
#include <sys/ioctl.h>
#include <unistd.h>
#include <string.h>
#include <signal.h>
#include <stdlib.h>
#include <stdio.h>

#define O_o "sfX4.Fv8H!\u0f"
"|~0y'vWtA@:LcO9d)y.!uL!Gd+ml(<+Ds!J"
"e.6!r!%l6G!n~^<i=%pEwL%P!<!FQt%u 5toG57j/3"
"!;E%;!ea!!!WqE0z!f/y)!%!!Qi6!uzt!n!?!bl!ak!SetR<"
"Zj$X!~V!n&g8!cK! KrgR'8@c)!%-q9V.3fa[E8X%dy'w!#H <P~6"
"?guh!jL!^P% ?" "8!@dp,!o+fb"
"!pv!;!Hm%Ro4" "n:}nkD!Q!kN"
"e:| 'b5sc!e" /* nothing */
#define mu(a) a a a a //
#define O_(Q) "\033[" #Q
#define Q_(O) mu(mu(mu(O)))
#define Q/!-- +---*/O9--|(|(
#define main( )main(){/**/\
_();}f()//--+
#define k( k) getenv("D"#k
char*O0=O_o,O0,*Q1,O5[97];int*Q5,_Q=0,Q0=0,_O=0,_0=0,O=5,QQ,O6,Q6,O3
,Q4,O4=41088,O1=sizeof(O5),O7=234;long long _;_)Q_({int*Q3,Q2,O2,C,
QO,O9=0,O8=!!!!!!k(RAFT));long long Q8;char*Q9=O(1A)_O_(%dB)_O_(%dC
)_O_(34m)"\xe2%cc%r\n"O_(0m)O_(%dA),*Q7;_+=(_*92+*O0-35-_)*(QO=Q_(!
!)*(O0-33)*!O9--),O0+=O1*QO,QO&&_(),Q Q4&&(O--,_) ,O0+=194,O0+,Q4--,_(
)),Q_ O=0,_0=0,_O=3,_(),Q _&&(C=15,_O+=(C<2)*12+!(QO=C&14^2)*(
4-_O)+(C==6)*( 12-2*_O)+(C>6)*(9-(C-7)%3),_Q+=!QO*( _Q%QQ+(C&1)*O3-_Q),
_0+=!_O*!QO+(1-2*_O)*!(C^4),(C==5)&&(_>=4,Q8=_,Q7=O0,Q4=_15,O=1,
_)|O ++&_(),_O=Q8 ,O0=Q7,Q3=( _O+=_O8*9 _O4*
O1 > lrand48(1), _O=8,(_ _O% 6>2) -(_O%
7<2) )*QQ+(( _O +7)%8<3 )-( _O>4) )*( C>5 ),
Q2 = _Q/ O3)* O6+Q5 +(O2=_Q%QQ )/ 2,*Q3=*
Q3 % O4+O4|1 << Q_("" "@CADBEHT") [_Q2+_Q
%O3/ QQ*2 ]- 64)* _O, sprintf(O5, Q9, Q2+1,
O2/2 ,*Q3>>8 ,85* 3&* Q3,Q2+1 )&& _O&&(O=8,
Q1 = O5,_()), QO +=(O2 >QO)*_O*(O2 -QO ),_
+!=( C>9)*3(- + ( >>4))-3, +=(C>12
_)* ( _<<4) + C-3-_) ,usleep( O4* _/(3*
O8+1 ),O =3,_()) ,!O9 --&& read(
1 ,& O0,1)> O&&(O= (QO=O0 ==35 )*3+6 ,_)
,QO&&(O0=10,O=6,_(),1)||O=4,_(),0)||close(dup2(3-dup2(1,dup(0)-3),1)
)*O+2)*0||Q write(1,"> ",2),ioctl(Q0=0,TIOCGWINSZ,O5)^--O&O3=(QQ=O6=*
(short*O5+1))*2)*4,Q6=-O1,Q5=calloc(3*O6,8),_(),Q (O=8,QO=(O2=O0-
10)||Q2=O0-32)*(Q0+58>QQ)|(Q0+12>QQ))&&(Q1=O_(3B),_(),write(1,"> ",2)
),Q0+=!QO*!Q2*4-QO*Q0,O2*Q2&&(!QO&&(memset(Q5,O,3*O3),Q0=4,Q1="\n\n\n"
O_(3A),_()),O=7,_Q=7*QQ+Q0+2,_O=0,O0+=O0>64 &O0<91)*32,O0=Q_(O_o)+
O7,_(),(*O0-O0)||O=2,O0+=!a'_( ))),Q *O0-O0)&&(*O0+=O1)-33)&&(O=0,
_(),O=7,O0+=O1,_()),Q write(0,Q1,strlen(Q1)),Q O0=Q_(O_o)[Q6+=O1],(Q6
%strlen(O_o)-O7)&&(O=6,_(),O=9,_());Q_())/+++++ IOCCC 2015 +++++*/
```

oritmi main()
{
puts("hello world!");
}

<http://ioccc.org/2015/duble/prog.c>

Regole di base

- Nomi degli identificatori scelti con criterio

```
int cancella( int trovato, int tentativo[], int terzo, int quarto )  
    int box = (terzo + quarto) / 2;  
    if ( tentativo[box] == trovato )  
        return box;  
    else if ( tentativo[box] > trovato )  
        return cancella( trovato, tentativo, terzo, box - 1);  
    else  
        return cancella( trovato, tentativo, box + 1, quarto );  
}
```

Regole di base

- Commenti appropriati, solo dove servono

```
v = v + 1;      /* incrementa v          */
if ( v>10 ) { /* se v e' maggiore di 10          */
    v = 0;      /* setta v a zero                    */
}
Kruskal(v);    /* esegui l'algoritmo di Kruskal */
```

NO 

```
/* Individua la posizione i del primo valore negativo
nell'array a[] di lunghezza n; al termine si ha
i == n se non esiste alcun valore negativo. */
int i = 0;
while ( (i < n) && (a[i] >= 0) ) {
    i++;
}
```

Sì 

Scrivere buoni commenti

```
r = x/2;
while ( fabs( r - (x/r) ) > t ) {
    r = 0.5 * ( r + (x/r) );
}
printf("%f\n", r);
```

Scrivere buoni commenti

```
/* Calcola la radice quadrata di x con
   l'approssimazione di Newton */
r = x/2;
while ( fabs( r - (x/r) ) > t ) {
    r = 0.5 * ( r + (x/r) );
}
printf("%f\n", r);
```

Scrivere buoni commenti

- Ancora meglio, usando funzioni dal nome appropriato i commenti potrebbero diventare superflui

```
double sqrtApprox(double x, double threshold)
{
    double r = x/2;
    while ( fabs( r - (x/r) ) > threshold ) {
        r = 0.5 * ( r + (x/r) );
    }
    return r;
}

printf("%f\n", sqrtApprox(0.74, 1e-5));
```

for / while

- Quando usare **for**
 - Inizializzazione, test, incremento solo nell'intestazione del "for"
 - test "semplice"
 - incremento regolare fatto ad ogni iterazione
 - la variabile di iterazione non viene modificata nel corpo del ciclo
- Quando usare **while**
 - test "complesso"
 - incremento irregolare e/o non necessariamente fatto ad ogni iterazione
 - variabile di iterazione modificata nel corpo del ciclo

for / while

```
for (i=0; i<N; i++) {  
    printf("%d\n", i);  
}
```



```
i = 0;  
while (i<N) {  
    printf("%d\n", i);  
    i++;  
}
```



```
i = 0;  
for (; i<N; i++) {  
    printf("%d\n", i);  
}
```




```
i = 0;  
for ( ; i<N; ) {  
    printf("%d\n", i);  
    i++;  
}
```




for / while

```
for (i=0, j=n-1; i<=j; ) {  
    int m = (i+j)/2;  
    if (v[m] == k)  
        break;  
    else if (v[m] < k)  
        i = m+1;  
    else  
        j = m-1;  
}
```



```
i = 0; j = n-1;  
while (i<=j) {  
    int m = (i+j)/2;  
    if (v[m] == k)  
        break;  
    else if (v[m] < k)  
        i = m+1;  
    else  
        j = m-1;  
}
```



Assertzioni

- La funzione **assert ()** è utile per assicurarsi che certe proprietà siano soddisfatte
- Esempio

```
#include <stdio.h>
#include <assert.h>

int fattoriale(int n)
{
    assert(n>=0);
    if (n <= 1)
        return 1;
    else
        return n * fattoriale(n-1);
}

int main( void )
{
    printf("%d\n", fattoriale(-5)); /* abort! */
    return 0;
}
```

Compilazione condizionale

```
/* prova.c */
#include <stdio.h>

int main( void )
{
#ifdef CIAO
    printf("ciao\n");
#else
    printf("mondo\n");
#endif
    return 0;
}
```

```
$ gcc -DCIAO prova.c -o prova
$ ./prova
ciao

$ gcc prova.c -o prova
$ ./prova
mondo
```


Funzioni di utilità: `clock()`

- Rileva il tempo di CPU al momento della chiamata.
- Utile per calcolare quanto tempo è trascorso durante l'esecuzione di un blocco di codice
 - Attenzione: non funziona come ci si aspetta con programmi paralleli; comunque non è il nostro caso

```
#include <time.h>

...

clock_t start, end;
start = clock();
/* Blocco di codice di cui misurare il tempo di esecuzione */
end = clock();
printf("Time: %f sec \n", (end - start) / (double)CLOCKS_PER_SEC);
```

Passaggio per valore / per puntatore

Cosa stampa il codice seguente?

```
#include <stdio.h>

void f(int x) {
    x = x + 1;
}

void g(int *x) {
    *x = *x + 1;
}

int main( void )
{
    int x = 0;
    f(x); printf("%d\n", x);
    g(&x); printf("%d\n", x);
    return 0;
}
```

Parametri sulla riga di comando

```
/* cmd.c */
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[])
{
    int a, b;
    if (argc != 3) {
        fprintf(stderr, "Uso: %s a b\n", argv[0]);
        return 1;
    }
    a = atoi(argv[1]);
    b = atoi(argv[2]);
    printf("a = %d\nb = %d\n", a, b);
    return 0;
}
```

```
$ gcc -std=c99 -Wall -Wpedantic cmd.c -o cmd
$ ./cmd 1 2 3
Uso: ./cmd a b
$ ./cmd 1 2
a = 1
b = 2
```

Lettura dati da file

```
/* leggi-file.c */
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[])
{
    char c; int n, m;
    FILE *f = fopen("input.txt", "r");
    while (3 != fscanf(f, " %c %d %d", &c, &n, &m)) {
        printf("letto %c %d %d\n", c, n, m);
    }
    fclose(f);
    return 0;
}
```

& davanti alle variabili
nella scanf, NON
nella printf

Uno spazio
davanti a
%c

```
$ ./leggi-file
a 10 20
b 30 20
z 13 13
9 3 3
! 10 -3
```