

Titolo unità didattica: Introduzione al linguaggio C

[03]

Titolo modulo : Input / Output in C

[05-C]

Operazioni elementari di lettura da tastiera e visualizzazione in C

Argomenti trattati:

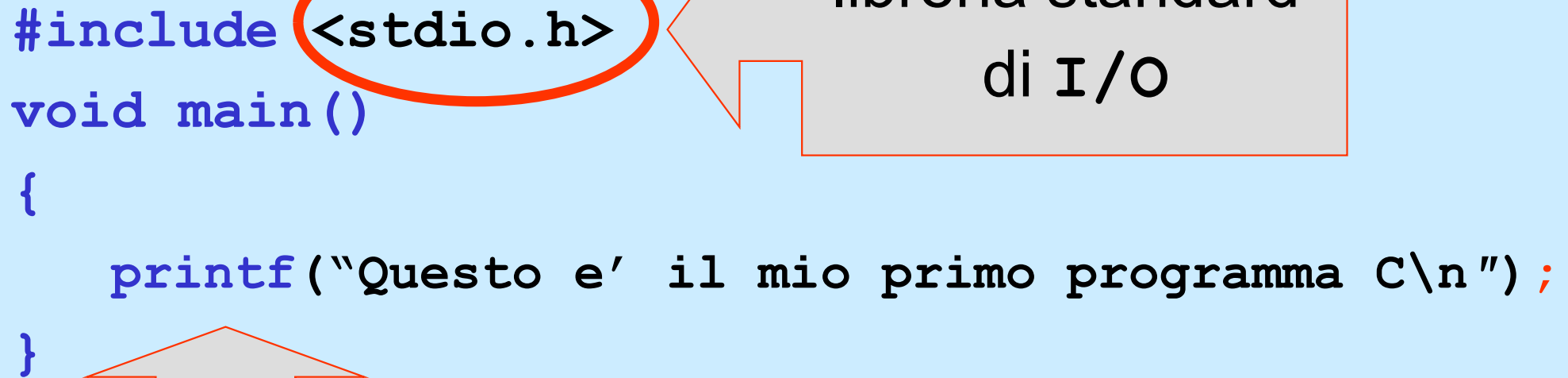
- ✓ operazione di visualizzazione in C: **printf**
- ✓ codici di formato in C
- ✓ operazione di lettura da tastiera in C: **scanf**
- ✓ I/O di caratteri in C

Prerequisiti richiesti: AP-03-04-C

operazioni di input e output (I/O) in C

Esempio

```
#include <stdio.h>
void main()
{
    printf("Questo e' il mio primo programma C\n");
}
```



libreria standard
di I/O

istruzione di output

```
Questo e' il mio primo programma C
Press any key to continue_
```

operazione di **output** in C

Esempio

```
#include <stdio.h>
void main()
{
    int eta;
    eta = 27;
    printf("valore di eta=%d/n",eta) ;
}
```

valore di eta=27

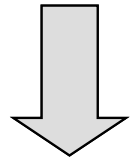
Press any key to continue_

operazione di **output** in C

```
printf("Questo e' il mio primo programma C\n")
```

```
printf("valore di eta=%d/n",eta) ;
```

```
printf(<stringa di controllo>,<variabili>) ;
```



produce la visualizzazione dei **valori** delle
<variabili>

la <stringa di controllo> contiene i **codici di formato** per i valori delle <variabili>

codici di formato in C

specificano le modalità di **conversione** tra la **rappresentazione interna** di un valore e la **rappresentazione** (notazione) **di visualizzazione**

%c

carattere

%d

int in notazione decimale

%f

float o **double** in notazione **[−]m.n**

%e

float o **double** in notazione **[−]m.nE[+-]xx**

%u

int in notazione decimale senza segno

%s

stringa

%o

notazione **ottale** senza segno

%x

notazione **esadecimale** senza segno

codici di formato in C

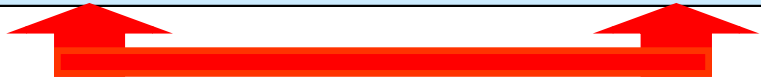
il **numero di codici di formato** nella `<stringa di controllo>` deve essere **uguale** al **numero di variabili** in `<variabili>`, cioè deve essere **uguale** al **numero di valori** da visualizzare

la **corrispondenza** tra **codice di formato** nella `<stringa di controllo>` e **relativa variabile** in `<variabili>` è per **posto**

la **posizione** nella `<stringa di controllo>` di un **codice di formato** indica la **posizione** dove viene visualizzato nella riga dello schermo il corrispondente valore della `<variabile>`

Esempio

```
int miglia, km;  
miglia = 1534;  
km = 97;  
printf("%d\n %d\n", miglia, km) ;
```



1534
97
_

```
float lun_maratona = 42.195F;  
printf("La maratona e' lunga %f chilometri\n",  
lun_maratona) ;
```

La maratona e' lunga 42.195000 chilometri
_

codici di formato in C

forma generale

%

— n

. m

carattere di conversione

%

carattere di inizio codice di formato

—

allineamento a sinistra

n

ampiezza del campo di visualizzazione

.

separatore

m

per una **stringa** massimo numero di caratteri da visualizzare
per un **int** massimo numero di cifre da visualizzare
per un **float** massimo numero di cifre da visualizzare dopo il
punto frazionario

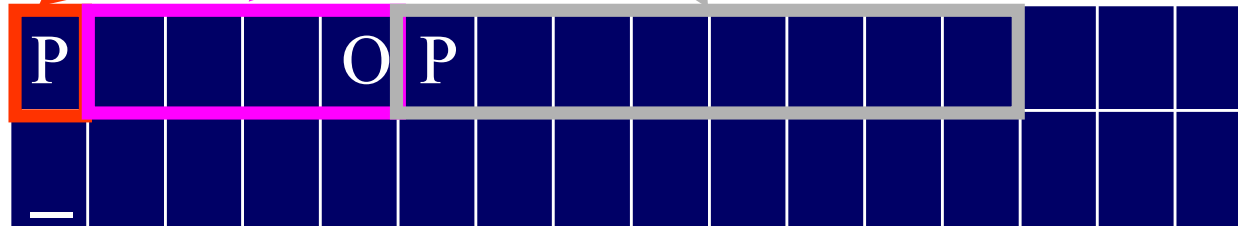
Esempio

```
char c1, c2;
```

```
c1 = 'P';
```

```
c2 = '0';
```

```
printf("%c%4c%-8c\n", c1, c2, c1);
```



Esempio

```
float x = 12.345678F;  
printf("%16.7e\n", x);
```



```
double xx = 12.34567890123456;  
printf("%22.15e\n%22.15f\n", xx, xx);
```

1.234567890123456e+001

12.345678901234560

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    int x;
    printf("Inserire un intero: ");
    scanf ("%d", &x);
    printf("valore inserito = %d\n", x);
}
```

istruzione di input

premere sul tasto

```
Inserire un intero: 21
valore inserito = 21
_
```

o)

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    int x,y;
    printf("Inserire due interi: ");
    scanf ("%d%d", &x,&y) ;
    printf ("primo valore inserito = %d\n
           secondo valore inserito = %d\n",x,y) ;
}
```

```
Inserire due interi: 321 654
primo valore inserito = 321
secondo valore inserito = 654
```

premendo sul tasto
(Invio)

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    int x,y;
    printf("Inserire due interi: ");
    scanf ("%d%d", &x,&y) ;
    printf ("primo valore inserito = %d\n",x) ;
    printf ("secondo valore inserito = %d\n",y) ;
}
```

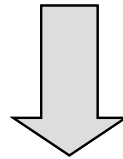
```
Inserire due interi: 321 654
primo valore inserito = 321
secondo valore inserito = 654
```

operazione di **input** in C

```
scanf ("%d", &x) ;
```

```
scanf ("%d%d", &x, &y) ;
```

```
scanf (<stringa di controllo>, <indirizzi variabili>) ;
```



sospende l'**esecuzione** del programma e resta in attesa dell'immissione da tastiera dei **valori** delle
<variabili>

la <stringa di controllo> contiene i **codici di formato** per i valori delle <variabili>

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    float x, y;
    printf("Inserire due numeri reali:\n");
    scanf("%f%f", &x, &y);
    printf("  x =%f\n y =%f\n", x, y);
}
```

Inserire due numeri reali :

1.0

2.45

x = 1.000000

y = 2.450000

—

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    float x, y;
    printf("Inserire due numeri reali:\n");
    scanf("%f%f", &x, &y);
    printf("  x =%f\n y =%f\n", x, y);
}
```

Inserire due numeri reali :

54.3257

256.67543E2

x = 54.325699

y = 25667.542969

—

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    float x, y;
    printf("Inserire due numeri reali:\n");
    scanf("%f%f", &x, &y);
    printf("  x =%20.3f\n  y =%12.5f\n", x, y);
}
```

Inserire due numeri reali :

54.3257

256.67543E2

x = 54.326

y = 25667.54297

—

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    float x, y;
    printf("Inserire due numeri reali:\n");
    scanf("%f%f", &x, &y);
    printf("  x =%15.7e\n  y =%15.7e\n", x, y);
}
```

Inserire due numeri reali :

54.3257

256.67543E2

x = 5.4325699e+001

y = 2.5667543e+004

—

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    int x;
    char y;
    printf("Inserire un intero e un carattere: ");
    scanf ("%d%c", &x, &y) ;
    printf ("x = %d e y = %c \n", x, y) ;
}
```

```
Inserire un intero e un carattere: 21A
x = 21 e y = A
```

```
_
```

operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    char x,y;
    printf("Inserire due caratteri: ");
    scanf ("%c%c",&x,&y);
    printf ("x = %c e y = %c \n", x, y);
}
```

Inserire due caratteri: GH

x =

e y = G

—



operazione di **input** in C

Esempio

```
#include <stdio.h>
void main()
{
    char x,y;
    printf("Inserire due caratteri: ");
    fflush(stdin);
    scanf ("%c%c", &x, &y);
    printf ("x = %c e y = %c \n", x, y);
}
```

Inserire due caratteri: GH

x = G e y = H

—

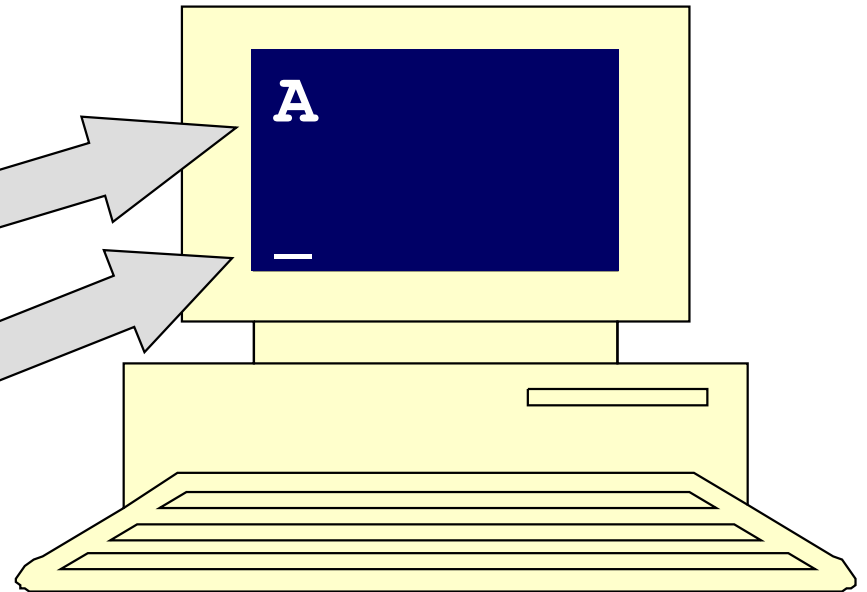
operazione di **output** di un singolo carattere

```
putchar(c) ;
```

visualizzazione del valore di
tipo **char** della variabile **c**

Esempio

```
char c = 'A' ;  
putchar(c) ;  
putchar("\n") ;
```



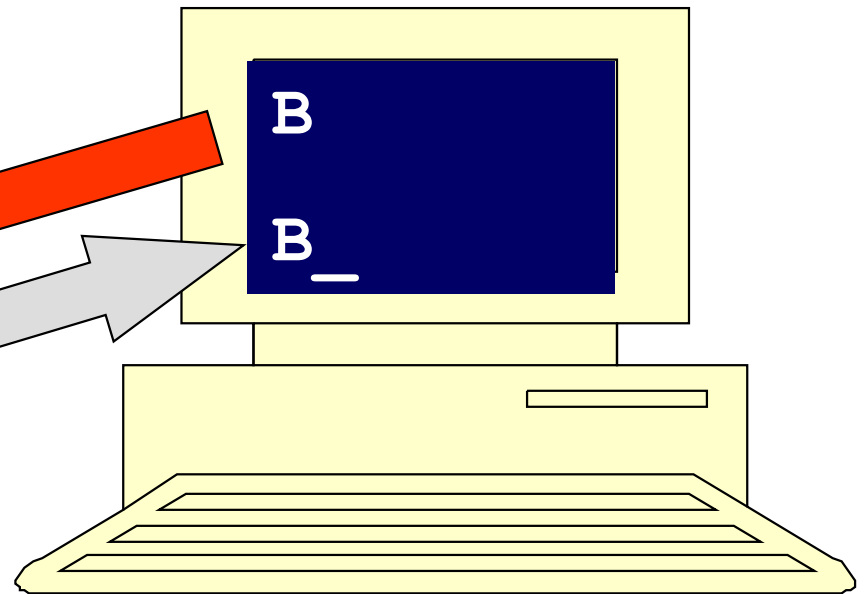
operazione di **input** di un singolo carattere

```
c = getchar() ;
```

lettura da tastiera del valore
di tipo **char** della variabile **c**

Esempio

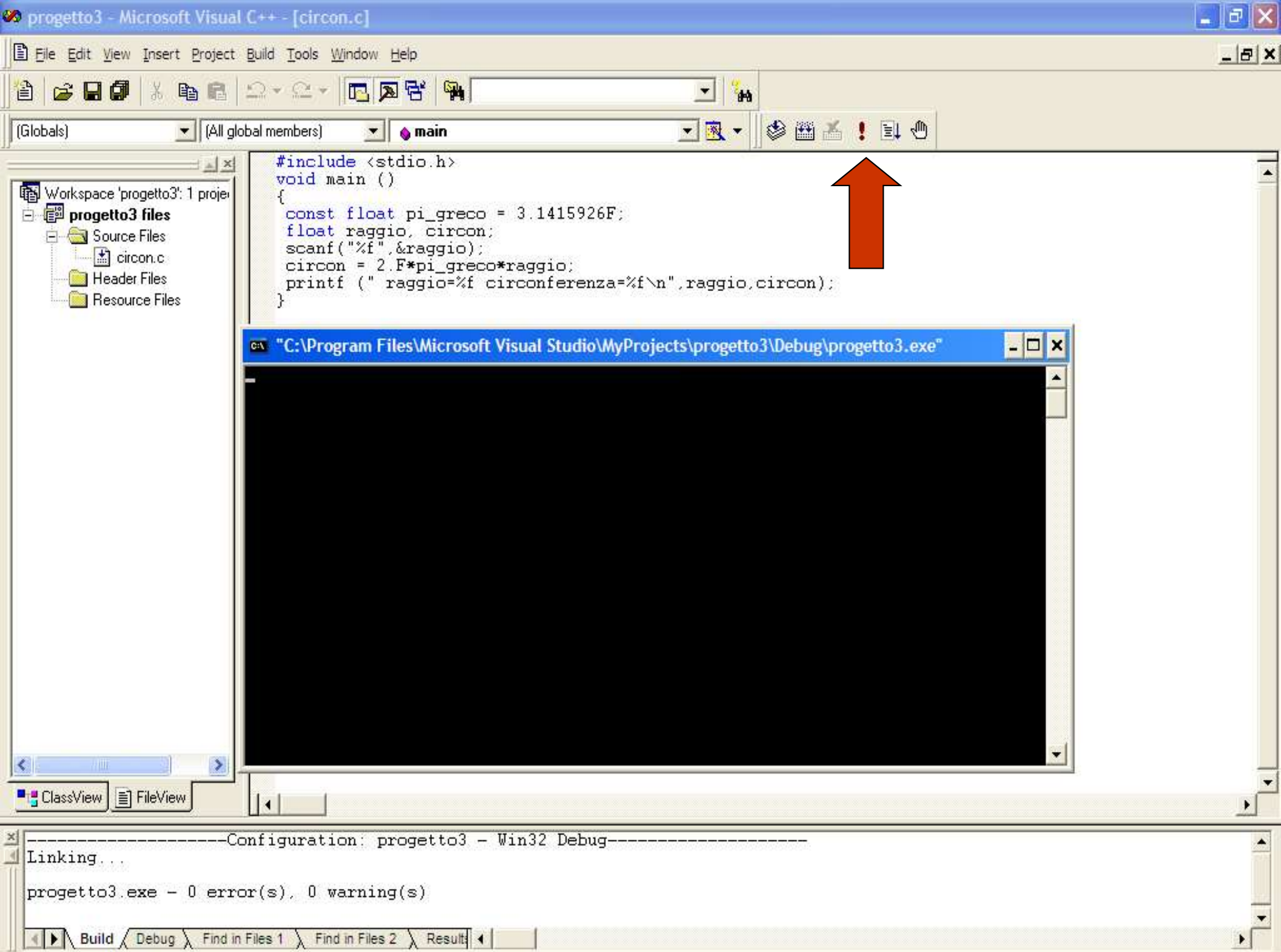
```
char c ;  
c = getchar() ;  
putchar(c) ;
```

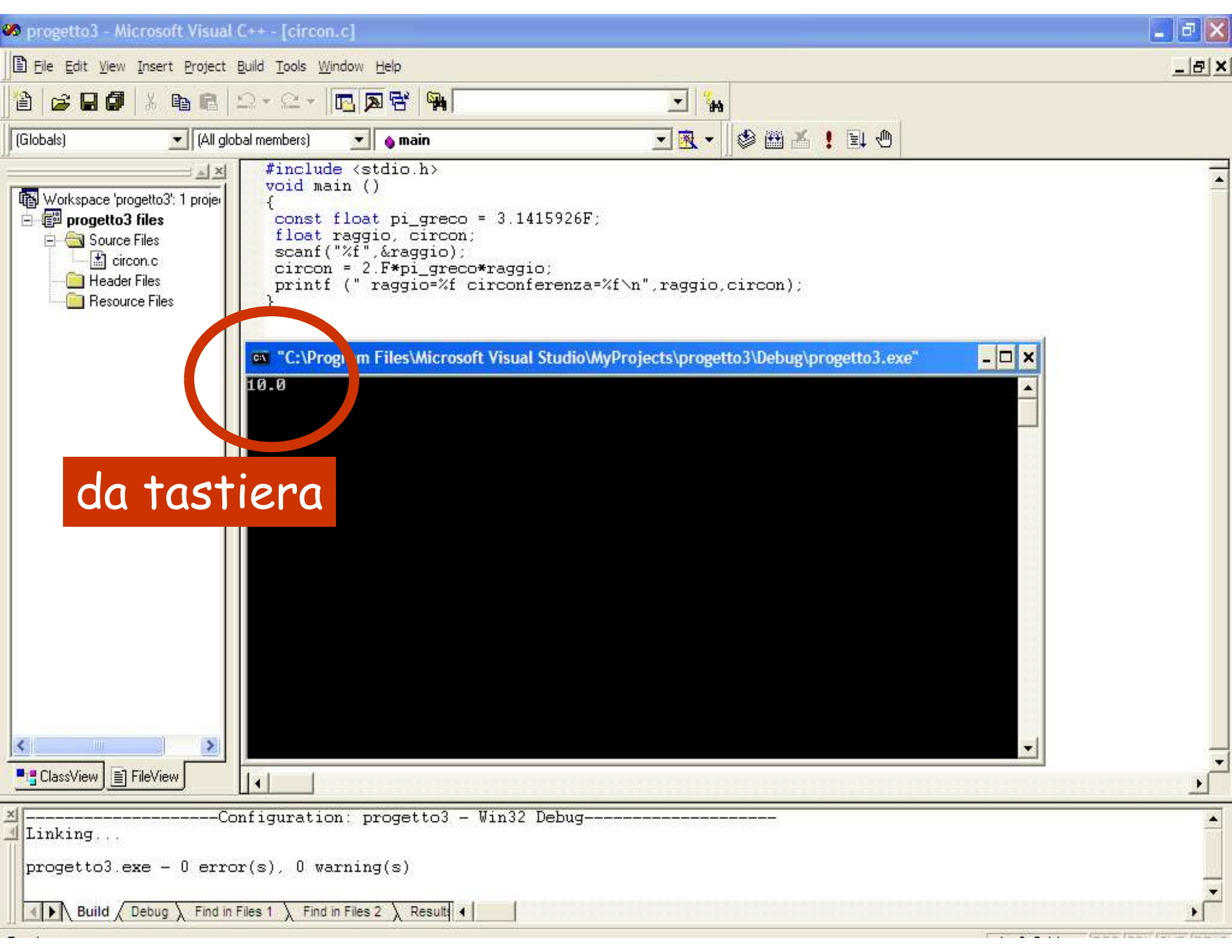


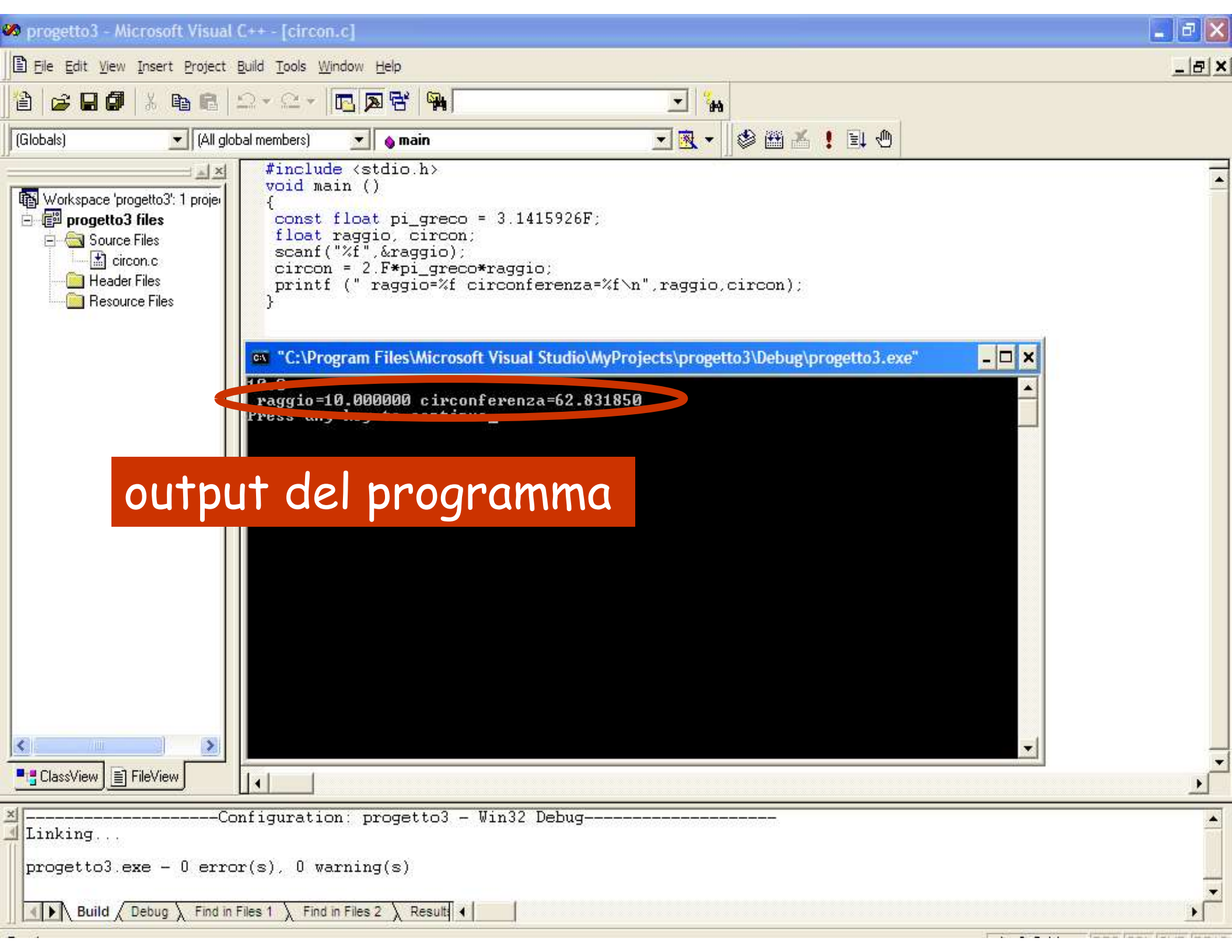
Esempio

calcolo e visualizzazione della circonferenza di un cerchio, dato (**lettura da tastiera**) il suo raggio

```
#include <stdio.h>
void main ()
{
    const float pi_greco = 3.1415926F;
    float raggio, circon;
    scanf ("%f", &raggio);
    circon = 2.F*pi_greco*raggio;
    printf ("raggio=%f  circonferenza=%f\n",
           raggio, circon);
}
```







output del programma

Esempio

calcolo e visualizzazione dell'area di un rettangolo,
date (**lettura da tastiera**) la sua base e la sua altezza

```
#include <stdio.h>
void main()
{
    int base, altezza, area;
    printf("Immettere la base del rettangolo (int): ");
    scanf("%d", &base);
    printf("\nImmettere l'altezza del rettangolo (int): ");
    scanf("%d", &altezza);
    area = base * altezza;
    printf("\n Area del rettangolo (base=%d,altezza=%d):%d\n",
        base, altezza, area);
}
```

Immettere la base del rettangolo (int): 5

Immettere l'altezza del rettangolo (int): 3

Area del rettangolo (base= 5, altezza= 3):15