

```
#include <stdio.h>

int main (int argc, char * argv[])
{
    int i = 10;
    float f = 30.455;
    printf ("I risultati sono %d e %6.2f\n", i, f);
}
```

- a. "I risultati sono %d e %6.2f, 10, 30.455"
- b. "I risultati sono 10 e 30.46"
- c. "I risultati sono 10 e 30.455"

%d valore decimale di una variabile intera

%6.2f valore di una variabile reale stampata in uno spazio di **6** caratteri, usando **2** cifre decimali

Risposte corrette = 68 %

Considerata la funzione C riportata in seguito, qual e' il risultato dell'invocazione "funz(8)"?

```
int funz (int p)
{
    int i, s;
    for (i = 0, s = 0; i < p; i++)
        if ((i % 2) == 0)
            s += i;
    return s;
}
```

"%" = operatore di "modulo"
(resto della divisione)

- a. 12
- b. 10
- c. 45

"i % 2" = resto della divisione di i per 2

L'effetto della funzione è la somma di tutti i numeri **pari** tra 0 e p-1.
funz (8) = 0 + 2 + 4 + 6 = 12

Risposte corrette = 50 %

Considerata la funzione C riportata in seguito, qual e' il risultato dell'invocazione "funz(30)"?

```
int funz (int p)
{
    int i, s;
    for (i = 0, s = 0; i < p; i++)
    {
        if (i >= 8)
            break;
        if ((i % 2) == 0)
            s += i;
    }
    return s;
}
```

- a. 210
- b. 12
- c. 45

break = interrompe l'esecuzione di un "blocco" di tipo "for", "while", "do/while", "case".

Il "for" termina appena **i** diventa uguale a (o maggiore di) 8. Questo, se $p \geq 8$ equivale a scrivere:

```
for (i = 0, s = 0; i < 8; i++)
```

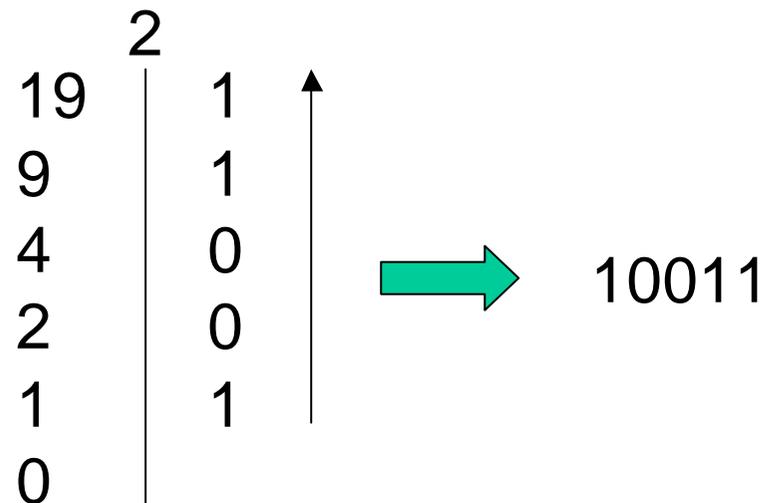
Risposte corrette = 48 %

Qual e' la rappresentazione in binario del numero (decimale) 19?

a. 11001

b. 10000

c. 10011



Risposte corrette = 90 %

Cosa rappresenta la dicitura di seguito?

```
int f (char * s, int i);
```

- a. la dichiarazione del prototipo della funzione “f”
- b. la dichiarazione dell’implementazione della funzione “f”
- c. l’invocazione della funzione “f”

Dichiarazione dell’implementazione:

```
int f (char * s, int i)
{
    // codice della funzione
}
```

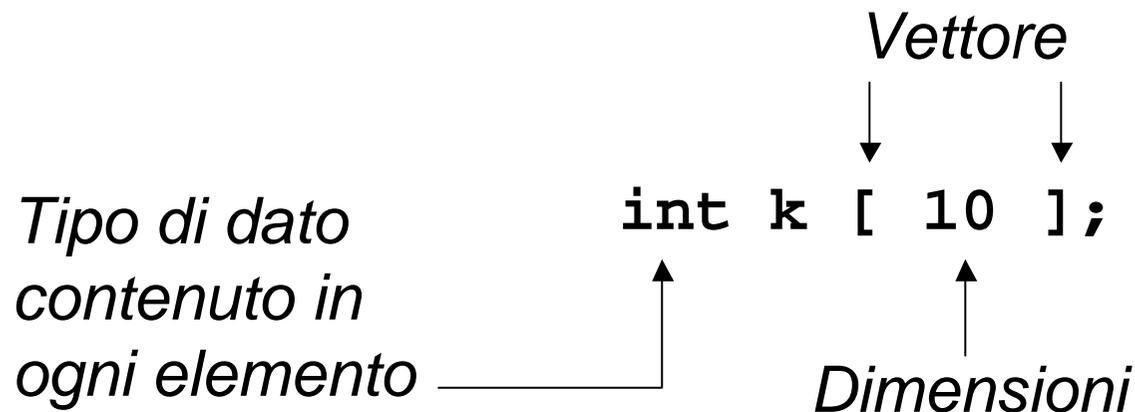
Invocazione:

```
char stringa[20];
int j, risultato;
...
risultato = f (stringa, j)
```

Risposte corrette = 56 %

Cosa significa la dichiarazione C “int k[10]”?

- a. che **k** e' un intero
- b. che **k** e' un vettore di interi di dimensioni non note
- c. che **k** e' un vettore di interi di dimensione uguale a 10



Risposte corrette = 90 %

Cosa produce l'esecuzione del seguente codice C?

```
int k[20];

int f (void)
{
    int i;
    for (i = 0; i < 100;i++)
        k [i] = i * 2;
}
```

- a. riempie il vettore k con i valori 0, 2, 4, 6, etc. fino a 198
- b. va in crash
- c. non produce nulla

Apparentemente l'effetto è "a", ma il vettore è dimensionato a 20, quindi non appena **i = 20** il programma va in crash!

Risposte corrette = 29 % !!!!

A cosa serve la funzione C “scanf”?

- a. a scandire una stringa
- b. a scandire un file
- c. a richiedere all'utente l'inserimento di un valore da tastiera

```
int scanf (char * fmt, <<puntatori alle variabili>>)
```

```
Inserimento di un intero:  
int i;  
...  
scanf ("%d", &i);
```

```
Inserimento di un carattere:  
char ch;  
...  
scanf ("%c", &ch);
```

```
Inserimento di una stringa:  
char str[20];  
...  
scanf ("%s", str);
```

Risposte corrette = 82 %

Quale delle seguenti righe di programma C e' (sintatticamente o semanticamente) errata?

```
a. int i; scanf ("%d", i);  
b. int i; scanf ("%d", &i);  
c. char s[100]; scanf ("%s", s);
```

Errore semantico,
Occorre passare un puntatore

Risposte corrette = 60 %

A cosa serve la funzione "strcmp"?

- a. a creare una stringa
- b. a confrontare due stringhe
- c. a cancellare una stringa

```
int strcmp (char * s1, char * s2);
```

= 0	s1 uguale a s2
> 0	s1 maggiore di s2
< 0	s1 minore di s2

Risposte corrette = 84 %