

## Test di Verifica AP-06-07-Test

- 1) Si consideri la function C `int somma_naturali(int n)`. La seguente chiamata di function è corretta:

```
int a;  
a = somma_naturali(22+5);
```

- V
- F

- 2) Si consideri la function C `int somma_naturali(int n)`. La seguente chiamata di function è corretta:

```
int n=4;  
1+2+3+4 = somma_naturali(n);
```

- V
- F

- 3) Quale delle seguenti function C calcola e restituisce in modo corretto la somma dei quadrati dei reciproci dei primi n numeri naturali?

- ```
float somma_quadr_rec(int n)  
{  
    float s;  
    int i;  
    s = 0.0F;  
    for (i = 1; i <= n; i++)  
        s = s + 1/(i*i);  
    return s;  
}
```
- ```
float somma_quadr_rec(int n)  
{  
    float s;  
    int i;  
    s = 0.0F;  
    for (i = 1; i <= n; i++)  
        s = (float)(s + 1/(i*i));  
    return s;  
}
```
- ```
float somma_quadr_rec(int n)  
{  
    float s;  
    int i;
```

```

s = 0.0F;
for (i = 1; i <= n; i++)
    s = s + 1/i*(float)i;
return s;
}

```

```

 float somma_quadr_rec(int n)
{
    float s;
    int i;
    s = 0.0F;
    for (i = 1; i <= n; i++)
        s = s + 1.0F/(i*i);
    return s;
}

```

4) Si consideri il seguente prototipo di function C:

```
float fun(float ,int );
```

Quale delle seguenti chiamate è corretta?

- float x,z=1.0F;  
x = fun(float z,0);
- float x,z=1.0F;  
x = fun(3+3,z);
- float x,z=1.0F;  
fun(z,0);
- float x,z=1.0F;  
z = fun(2\*z,0);

5) Si consideri la function C `int fattorialeI(int n)`. La seguente chiamata di function è corretta:

```
int n=10;
1*2*3*...*n = fattorialeI(n);
```

- V
- F

6) Si consideri la function C `int fattorialeI(int n)`. La seguente chiamata di function è corretta:

```
int x,n=10;
x = fattorialeI(n*n);
```

- V
- F

7) Si consideri la function C `int fattorialeI(int n)`. La seguente chiamata di function è corretta:

```
int n=10;
n = fattorialeI(n);
```

- V
- F

8) Quale delle seguenti chiamate alla function `fattorialeF` consente che il valore calcolato del fattoriale sia effettivamente associato alla variabile `a`?

- `int a;`  
`a = fattorialeF(10);`
- `int a,n=10;`  
`a = fattorialeF(n);`
- `int n=10;`  
`float a;`  
`a = fattorialeF(10);`
- `int n=10;`  
`char a;`  
`a = fattorialeF(10);`

9) Si consideri la function C `int fattorialeI(int n)`. Quale delle seguenti espressioni consente di calcolare il prodotto  $5*6*7*8*9*10$  ??

- `int a;`  
`a = fattorialeI(10)-fattorialeI(4);`
- `int a;`  
`a = fattorialeI(10)/fattorialeI(4);`
- `int a;`  
`a = fattorialeI(10)/fattorialeI(5);`
- `int a;`  
`a = (float)fattorialeI(10)/(float)fattorialeI(4);`

10) In C esiste una istruzione `break`, che quando viene eseguita provoca l'uscita dal ciclo in cui si trova. Dire se la seguente porzione di programma C è una corretta implementazione C dell'algoritmo di ricerca sequenziale di dati da tastiera:

```

for(i=1; i<=n; i++) {
    printf("inserire %d-simo carattere:",i);
    fflush(stdin);
    scanf("%c",&dato_letto);
    if(dato_letto == chiave)
        break ;
}
if(i == n)
    printf("chiave non trovata");
else
    printf("chiave trovata");

```

- V
- F

## Problemi aperti

1. Sviluppare una versione `double fattorialeD(int n)` della function per il fattoriale. Confrontando i valori del fattoriale calcolati da tale function e dalla function `float fattorialeF(int n)`, sviluppata nel modulo AP-06-06-C, determinare fino a quale valore di `n` la function `fattorialeF` è in grado di calcolare il valore corretto del fattoriale.

---

Modulo 3 (quiz vero-falso)

- 1) Gli elementi di un array vengono allocati in memoria in maniera contigua ??? (VERO)
- 2) Il nome di un array rappresenta l'indirizzo dell'ultima componente del vettore ??? (FALSO)
- 3) L'uso degli indici per l'accesso alle componenti del vettore rende il programma più efficiente, rispetto all'uso dei puntatori (dereferenziazione indirizzi elementi) ??? (FALSO)
- 4) Qualunque operazione eseguita usando un array e un indice può essere implementata usando i puntatori ??? (VERO)
- 5) In genere, valori di tipo intero occupano in memoria 4 Bytes ??? (VERO)
- 6) L'operatore `sizeof` restituisce la dimensione dell'array ??? (FALSO)
- 7) E' corretto sostenere che gli array sono puntatori costanti ??? (VERO)
- 8) Per eseguire un algoritmo di ricerca sequenziale l'array iniziale deve essere ordinato ??? (FALSO)
- 9) Per eseguire un algoritmo per verificare l'uguaglianza tra due array è necessario che la dimensione degli stessi sia uguale??? (VERO)
- 10) Per eseguire un algoritmo di fusione tra 2 array è necessario che gli elementi presenti negli array siano tutti positivi (> 0) (FALSO)

#### Modulo 4 (quiz vero-falso)

- 1) Un array è un insieme di variabili dello stesso tipo ??? (VERO)
- 2) La dichiarazione di un array è utilizzata dal compilatore per riservare in memoria spazio sufficiente all'array ??? (VERO)
- 3) In un array di stringhe l'ultimo elemento è sempre il carattere '\0' ?? (VERO)
- 4) Sia dato un array  $A=[1,2,3,4]$  di 4 elementi.  
Per accedere all'ultimo elemento dell'array è corretta la scrittura :  $A[4]$  ??? (FALSO)
- 5) Dopo aver dichiarato un array è sempre possibile modificare la dimensione dello stesso?? (FALSO)
- 6) Gli array in linguaggio C vengono sempre passati alle function per riferimento ??? (FALSO)

## Esercizio

Sia dichiarato l'array di interi A di 4 elementi:  $A = [1,2,3,4]$

Sia dichiarato un puntatore `int *p;`

Verificare se le seguenti affermazioni/assegnazioni sono vere o false :

- 1) `A[3] = 4` ?? (VERO)
- 2) `*A+2 = 2` ?? (FALSO) -> 3
- 3) `A[4] = 4` ?? (FALSO) -> Fuori dalla dimensione dell'array
- 4) `*a+(sizeof(int)) = 4` (FALSO) -> Fuori dalla dimensione dell'array
- 5) `*a+(sizeof(int)-3) = 2` (VERO)
- 6) `&a[0] = 1` (FALSO) -> Indirizzo di memoria del primo elemento dell'array
- 7) `p = *a;` (Assegnazione errata)
- 8) `p = &a; p[0] = 1;` (VERO)
- 9) `A[0] + &p[0] = 2` ? (FALSO)
- 10) `*a[1] = 2;` (Assegnazione errata)