

**Esercizio 1** (punti 14)

Scrivere l'algoritmo ed il codice C di un programma che legge da input un carattere e richiama una procedura che inserisce il carattere appena letto all'interno di un array. La procedura prevede di ricevere un array **ordinato alfabeticamente** e lo deve lasciare ordinato a seguito dell'inserimento, quindi deve memorizzare il nuovo carattere nella posizione corretta, facendo scorrere i caratteri più grandi nelle posizioni successive.

**Esercizio 2** (punti 4)

In che cosa si differenziano la funzione e la procedura riportati di seguito?  
Quale è la sintassi corretta per chiamarle entrambe?

<pre>float funzione(char a, float b) {     if (a=='p')         return b/2.0;     else return b; }</pre>	<pre>void procedura(char a, float b, float *c) {     if (a=='p')         *c = b/2.0;     else *c = b; }</pre>
---	---

**Esercizio 3** (punti 5)

Scrivere il codice della funzione moltiplica, di interfaccia nota, che calcola il valore al passo n della produttoria riportata di seguito, data la costante a.

int **moltiplica**(int a, int n);

$$\prod_{j=1}^n (a * j - a + j)$$

**Esercizio 4** (punti 8)

Cosa viene stampato dal programma, nell'ipotesi che i valori letti dalla scanf siano le ultime 4 cifre del numero della matricola? La risposta deve essere opportunamente motivata. Si dica inoltre se vengono infrante regole di visibilità e se la variabile i definita nell'area globale è visibile dalla procedura stampa.

<pre>#include&lt;stdio.h&gt; #define Dim 4 int i=0; int A[Dim]={1,2,3,4}; int vet[Dim];  void somma(int vet1[], int vet2[]); void stampa(int vet[]);  main(){     int j ;     for (j=0; j&lt;DIM; j++)         scanf("%d", &amp;vet[ j ])     stampa(A);     somma(A,vet);     stampa(vet);     printf("%d",vet[i]); }</pre>	<pre>void somma(int vet1[Dim], int vet2[Dim]){     int i;     for(i=0;i&lt;Dim;i++) vet2[i]=vet2[i]         +vet1[i]; }  void stampa(int vet[]){     int i;     printf("Vettore:\n");     for(i=0;i&lt;Dim;i++) printf("%d ",vet         [i]);     printf("\n"); }</pre>
--	--