



Qualcosa da ricordare...

Puntatori a una variabile

```
int main(void) {
    int a = 9;
    printf("%d\n", a ); // 9 = valore di a
    printf("%p\n", &a); // 0x16fdff2dc = indirizzo di a
}
```

```
int a = 9, *punt_a;
// il puntatore *punt_a va inizializzato con l'indirizzo di a
punt_a = &a;
printf("%d \n", *punt_a); // 9 = valore di a
printf("%p \n", punt_a); // 0x16fdff2dc = indirizzo di a
*punt_a = *punt_a + *punt_a;
printf("%d \n", a); // 18 = valore di a
printf("%d \n", *punt_a); // 18 = valore di a
printf("%p \n", punt_a); // 0x16fdff2dc = indirizzo di a
```

Puntatori a un array

```
int a[10] = {0,1,2,3,4,5,6,7,8,9}, *punt_a;  
// il puntatore *punt_a va inizializzato con l'indirizzo di a[]  
punt_a = a;  
printf("%p \n", punt_a); // 0x16fdff2b0 = indirizzo di a[]  
printf("%p \n", a);     // 0x16fdff2b0  
printf("%p \n", &a[0]); // 0x16fdff2b0
```

Passiamo un puntatore ad un array a una funzione.



Esercizi sugli array

Titolo unità didattica: Strutture dati: array

[07]

Titolo modulo : Function in C per problemi di base con array

– parte 1

[09-C]

Sviluppo di function in C per algoritmi di base per array 1D ed esempi di utilizzo

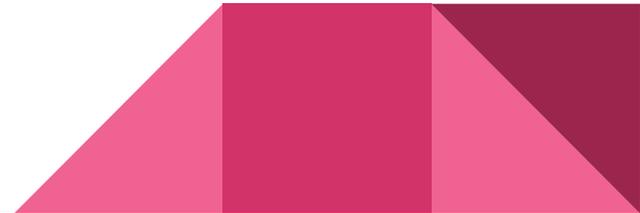
Argomenti trattati:

- ✓ function in C per somma, media e varianza
- ✓ function in C per la somma cumulativa
- ✓ function in C per il massimo, il minimo e il massimo/minimo

Prerequisiti richiesti: AP-05-03-C, AP-07-02-T, AP-07-08-C

Esercizio 1)

Realizzare un main C che calcola la somma degli elementi di un array

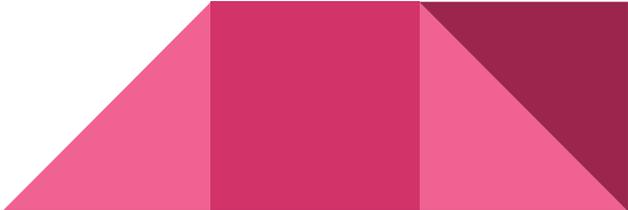


Esercizio 2)

Realizzare un main C che calcola la somma degli elementi di un array, utilizzando una funzione.
Implementa entrambi i prototipi.

```
int somma_arrayI (int a[], int n) // standard
```

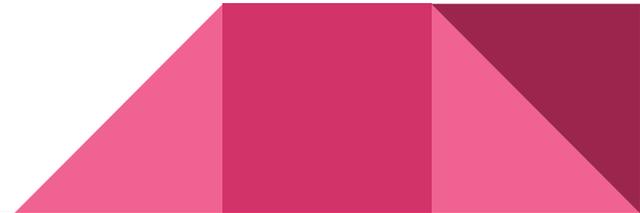
```
int somma_arrayI (int *a, int n) // puntatori
```



```
2 #include <stdlib.h>
3
4 int somma_standard(int a[], int n); //
5 int somma_puntatori(int *a, int n);
6
7
8 int main(void) {
9     int a[10] = {0,1,2,3,4,5,6,7,8,9};
10    int i = 0, somma = 0;
11    for(i = 0 ; i<10; i++){
12        somma += a[i];
13    }
14    printf(" somma main = %d \n", somma);
15    printf(" somma standard = %d \n", somma_standard(a, 10));
16    printf(" somma puntatori = %d \n", somma_puntatori(a, 10));
17
18    return 0;
19 }
20
21
22 int somma_standard(int a[], int n){
23     int i = 0, somma = 0;
24     for(i = 0 ; i<10; i++){
25         somma += a[i];
26     }
27     return somma;
28 }
29
30 int somma_puntatori(int *a, int n){
31     int i = 0, somma = 0;
32     for(i = 0 ; i<10; i++){
33         somma += a[i];
34     }
35     return somma;
36 }
```

Esercizio 3)

Realizzare un main C che calcola la somma degli elementi di un array, utilizzando una procedura.



Esercizio 3)

```
int somma_standard(int a[], int n); //
int somma_puntatori(int *a, int n);
void somma_procedura(int *a, int n, int *somma);

int main(void) {
    int a[10] = {0,1,2,3,4,5,6,7,8,9};
    int i = 0, somma = 0;
    for(i = 0 ; i<10; i++){
        somma += a[i];
    }
    printf(" somma main = %d \n", somma);
    printf(" somma standard = %d \n", somma_standard(a, 10));
    printf(" somma puntatori = %d \n", somma_puntatori(a, 10));
    somma_procedura(a, 10, &somma);
    printf(" somma puntatori = %d \n", somma);

    return 0;
}

void somma_procedura(int a[], int n , int *somma){
    int i = 0;
    *somma = 0;
    for(i = 0 ; i<10; i++){
        *somma += a[i];
    }
}
```

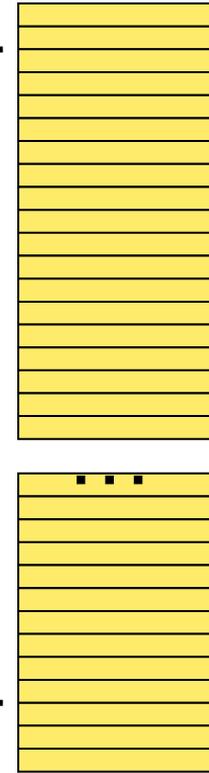
```
int a[100];
```

associa 100 celle
all'array **a**

```
printf("inserire il numero di elementi  
(<=100):");  
scanf("%d",&n_elem);  
for (i=0;i<n_elem;i++)  
    scanf("%d",&a[i]);
```

10

```
somma = somma_arrayI(a,n_elem);  
printf ("somma degli elementi: %d  
\n",somma);  
printf ("l'array e' \n");  
visualizza_aI(a,n_elem);  
}
```



vengono
utilizzate solo
le prime 10
celle

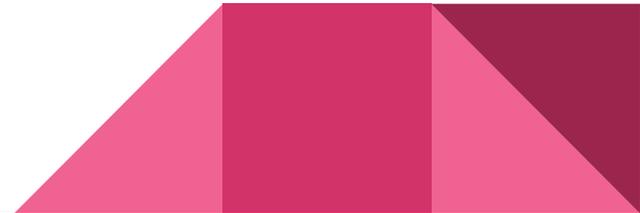
agiscono
solo sulle
prime 10
celle

Esercizio 3-b)

Realizzare un main C che calcola la somma degli elementi di un array, utilizzando una procedura.

Crea un array molto grande e inizializza con scanf solo le prima n posizioni.

Cambia la procedura in modo da memorizzare la somma nell'elemento (n+1)-esimo dell'array



```
void somma_procedura_11(int *a, int n);

int main(void) {
    int a[11] = {0,1,2,3,4,5,6,7,8,9};
    int i = 0, somma = 0;
    int somma_s, somma_p;

    for(i = 0 ; i<10; i++)
        somma += a[i];
    printf("somma main = %d \n", somma);
    somma_s = somma_standard(a, 10);
    printf("somma standard = %d \n", somma_s);
    somma_p = somma_puntatori(a,10);
    printf("somma puntatori = %d \n", somma_p);
    somma_procedura(a, 10, &somma);
    printf("somma procedura= %d \n", somma);
    somma_procedura_11(a, 10);
    printf("somma procedura = %d \n", a[10]);
    return 0;
}
```

```
void somma_procedura_11(int *b, int num){
    int i=0;
    b[10] = 0;
    for(i = 0 ; i<num; i++)
        b[10] += b[i];
}
```

Esercizio 4)

Realizzare una function C che calcola la media degli elementi di un array.



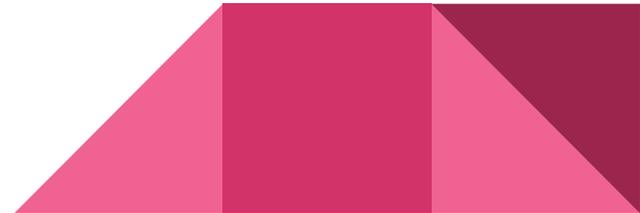
esercizio

realizzare una function C che calcola la media degli elementi di un array

```
/* media degli elementi di un array double
   - notazione standard */
double media_arrayD(double a[], int n)
{
    double s;
    int i;
    s = 0.0;
    for (i=0;i<n;i++)
        s = s + a[i];
    return s/n;
}
```

Esercizio 5)

Realizzare una function C che calcola la varianza degli elementi di un array.



Esercizio 5)

Realizzare una function C che calcola la varianza degli elementi di un array.



esercizio

realizzare una function C che calcola la varianza degli elementi di un array

```
/* varianza degli elementi di un array double
   - notazione standard */
double varianza_arrayD(double a[], int n)
{
    double s=0.0, media;
    int i;
    media = media_arrayD(a,n);
    for (i=0;i<n;i++)
        s = s + pow(a[i] - media,2);
    return s/(n-1);
}
```

esercizio

realizzare una function C che calcola l'array delle somme cumulative degli elementi di un array



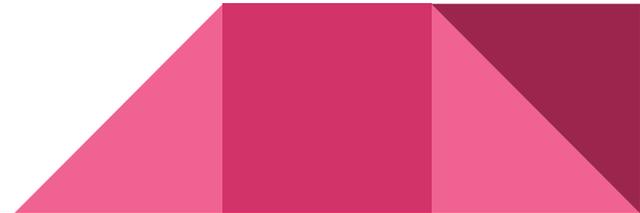
```
/* somma cumulativa degli elementi di un array double
- notazione standard
void somma_cumD(double a[], int n, double sum_cum[n] */*
{
    int i;
    sum_cum[0] = a[0];
    for (i=1; i<n; i++)
        sum_cum[i] = sum_cum[i-1] + a[i];
}
```

parametro di uscita

```
#include <stdio.h>
void somma_cumD(double [], int, double []);
void legge_da_tastiera_aD(double [], int );
void visualizza_aD (double [], int );
void main ()
{
    int n_elem;
    double vet[100],vet_somme_cum[100];
    printf("inserire il numero di elementi (<=100):");
    scanf("%d",&n_elem);
    legge_da_tastiera_aD(vet,n_elem);
    somma_cumD(vet,n_elem,vet_somme_cum);
    visualizza_aD(vet_somme_cum,n_elem);
}
void somma_cumD(double a[],int n,double sum_cum[])
{
```

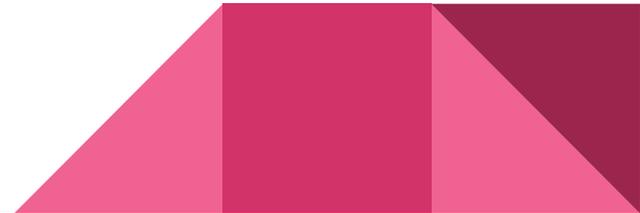
Esercizio 6)

Realizzare un main C che determina la somma cumulativa degli elementi di un array.



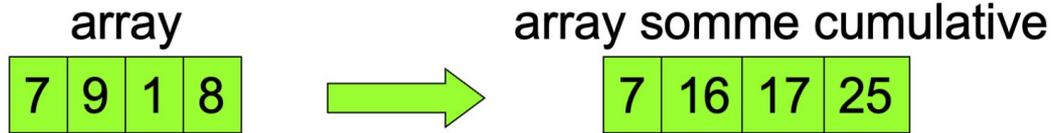
Esercizio 7)

Realizzare una function C che determina il massimo elemento di un array.



esercizio

realizzare una function C che calcola l'array delle somme cumulative degli elementi di un array



```
/* somma cumulativa degli elementi di un array double
   - notazione standard */
void somma_cumD(double a[],int n,double sum_cum[])
{
    int i;
    sum_cum[0] = a[0];
    for (i=1;i<n;i++)
        sum_cum[i] = sum_cum[i-1] + a[i];
}
```

parametro di uscita

esercizio

realizzare un main C che determina il massimo elemento di un array

```
#include <stdio.h>
void main ()
{
    int a[] = {2,15,1,-1,22,0,-2};
    int i,max;
    max = a[0];
    for (i=1;i<7;i++)
        if(a[i] > max)
            max = a[i];
    printf ("elemento massimo: %d \n",max);
    printf ("l'array e' \n");
    for (i=0;i<7;i++)
```

esercizio

realizzare una function C che determina il massimo elemento di un array

```
/* massimo tra gli elementi di un array int
   - notazione standard */
int massimo_arrayI (int a[], int n)
{
    int max, i;
    max = a[0];
    for (i=1;i<n;i++)
        if(a[i] > max)
            max = a[i];
    return max;
}
```

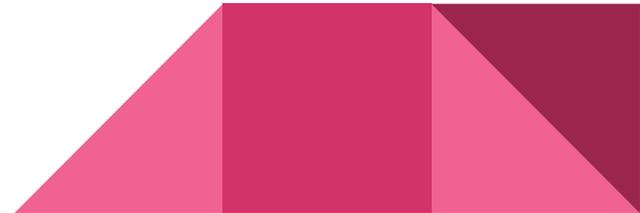
esercizio

realizzare un main che chiama la function C che determina il massimo elemento di un array

```
#include <stdio.h>
int massimo_arrayI(int [ ], int);
void legge_da_tastiera_aI(int [], int );
void visualizza_aI (int [], int );
void main ()
{
    int a[100],massimo,n_elem;
    printf("inserire il numero di elementi (<=100):");
    scanf("%d",&n_elem);
    legge_da_tastiera_aI(a,n_elem);
    massimo = massimo_arrayI(a,n_elem);
    printf ("il massimo tra gli elementi: %d \n",massimo);
    printf ("l'array e' \n");
    visualizza_aI(a,n_elem);
}
```

Esercizio 8)

Realizzare una function C che determina il minimo elemento di un array.



esercizio

realizzare una function C che determina il minimo elemento di un array

```
/* minimo tra gli elementi di un array int
   - notazione standard */
int minimo_arrayI (int a[], int n)
{
    int min, i;
    min = a[0];
    for (i=1;i<n;i++)
        if(a[i] < min)
            min = a[i];
    return min;
}
```

Esercizio 9)

Realizzare una procedura C che determina il massimo e il minimo elemento di un array.

```
void max_min_arrayI(int a[], int n, int *max, int *min)
```



esercizio

realizzare una function C che determina il massimo e il minimo elemento di un array

```
/* massimo e minimo tra gli elementi di un
array int - notazione standard */
void max_min_arrayI(int a[],int n,
                    int *max, int *min)
{
    int i;
    *max = a[0];
    *min = a[0];
    for (i=1;i<n;i++)
        if(a[i] > *max)
            *max = a[i];
```

esercizio

realizzare un main che chiama la function C che determina il massimo e il minimo elemento di un array

```
#include <stdio.h>
int max_min_arrayI(int [], int, int *, int *);
void legge_da_tastiera_aI(int [], int );
void visualizza_aI (int [], int );
void main ()
{
    int a[100], massimo, minimo, n_elem;
    printf("inserire il numero di elementi (<=100):");
    scanf("%d", &n_elem);
    legge_da_tastiera_aI(a, n_elem);
    max_min_arrayI(a, n_elem, &massimo, &minimo);
    printf ("il massimo tra gli elementi: %d \n", massimo);
    printf ("il minimo tra gli elementi: %d \n", minimo);
```