



# IEIM 2019-2020

## Esercitazione II *“Array”*

Alessandro A. Nacci

[alessandro.nacci@polimi.it](mailto:alessandro.nacci@polimi.it) - [www.alessandronacci.it](http://www.alessandronacci.it)



# Gli array

- Supponiamo di voler memorizzare i voti di 10 persone di una classe. L'unica soluzione che abbiamo al momento è creare 10 variabili

```
int voto_1 = 29;  
int voto_2 = 14;  
int voto_3 = 3;  
int voto_4 = 15;  
  
...  
int voto_9 = 17;  
int voto_10 = 21;
```



# Gli array

- Supponiamo di voler memorizzare i voti di 10 persone di una classe. L'unica soluzione che abbiamo al momento è creare 10 variabili

```
int voto_1 = 29;  
int voto_2 = 14;  
int voto_3 = 3;  
int voto_4 = 15;  
  
...  
int voto_9 = 17;  
int voto_10 = 21;
```

E se volessimo adesso  
memorizzarne 100?

**MOLTO SCOMODO!**

Dovrei dichiarare 100 variabili!



# Gli array

- Supponiamo di voler memorizzare i voti di 10 persone di una classe. L'unica soluzione che abbiamo al momento è creare 10 variabili

## Con Variabili

```
int voto_1 = 29;  
int voto_2 = 14;  
int voto_3 = 3;  
int voto_4 = 15;  
...  
int voto_9 = 17;  
int voto_10 = 21;
```



## Con Array

```
int voti[100];  
voto[0] = 29;  
voto[1] = 14;  
voto[2] = 3;  
...  
voto[8] = 17;  
voto[9] = 21;
```



```
es1.c x
1 #include <stdio.h>
2
3 int main()
4 {
5
6     int a1, a2, a3, a4, a5, a6, a7, a8, a9, a10;
7     a0 = 120;
8     a1 = 10;
9     a2 = 20;
10    a3 = 24;
11    a4 = 21;
12    a5 = 20;
13    a6 = 210;
14    a7 = 208;
15    a8 = 2670;
16    a9 = 230;
17
18    printf("%d", a0);
19    printf("%d", a1);
20    printf("%d", a2);
21    printf("%d", a3);
22    printf("%d", a4);
23    printf("%d", a5);
24    printf("%d", a6);
25    printf("%d", a7);
26    printf("%d", a8);
27    printf("%d", a9);
28
29
30
31 }
```

```
untitled
1 #include <stdio.h>
2
3 int main()
4 {
5
6     int v[10];
7     int cont = 0;
8
9     v[0] = 120;
10    v[1] = 10;
11    v[2] = 20;
12    v[3] = 24;
13    v[4] = 21;
14    v[5] = 20;
15    v[6] = 210;
16    v[7] = 208;
17    v[8] = 2670;
18    v[9] = 230;
19
20    do
21    {
22        printf("%d", v[cont]);
23        cont = cont + 1;
24    } while(cont < 10);
25
26 }
```



# Gli array

- Supponiamo di voler memorizzare i voti di 10 persone di una classe. L'unica soluzione che abbiamo al momento è creare 10 variabili

## Con Variabili

```
int voto_1 = 29;  
int voto_2 = 14;  
int voto_3 = 3;  
int voto_4 = 15;  
...  
int voto_9 = 17;  
int voto_10 = 21;
```



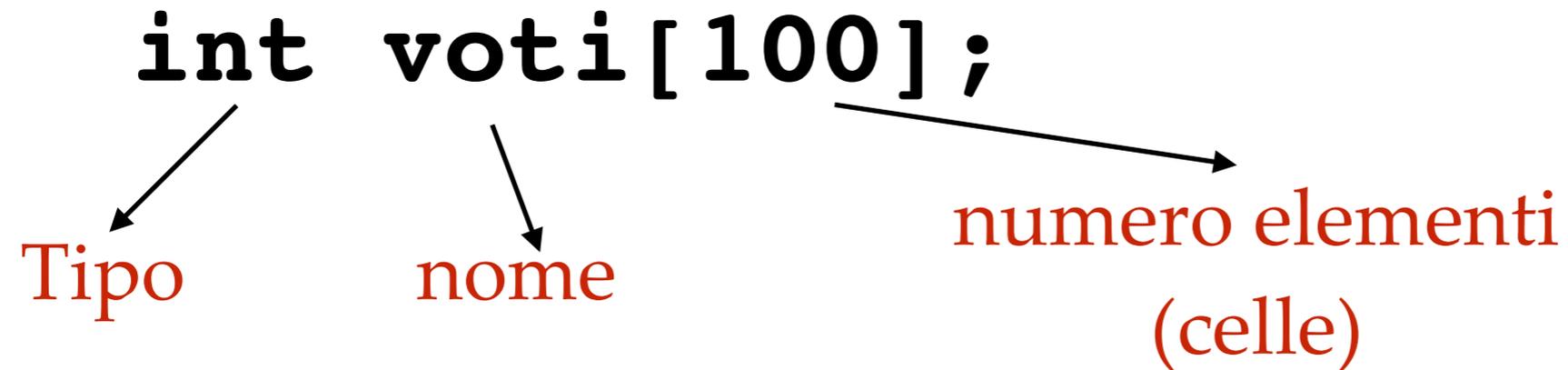
## Con Array

```
int voti[100];  
voto[0] = 29;  
voto[1] = 14;  
voto[2] = 3;  
...  
voto[8] = 17;  
voto[9] = 21;
```

La prima posizione è  
0 e non 1!!!



## Dichiarazione di un array:



## Utilizzo degli array:

**voto[3]=5;**





## Perché sono comodi? Perché posso iterare!

### Con Variabili

```
scanf("%d", &voto_1);  
scanf("%d", &voto_2);  
scanf("%d", &voto_3);  
scanf("%d", &voto_4);  
...  
scanf("%d", &voto_99);  
scanf("%d", &voto_199);
```



### Con Array

```
int voti[100];  
int i = 0;  
  
do{  
    scanf("%d", &voti[i]);  
    i = i + 1;  
} while(i<100);
```



## Ragioniamo sul ciclo....

### Do-While

```
int voti[100];
int i = 0;

do{
    scanf("%d", &voti[i]);
    i = i + 1;
} while(i<100);
```

### Ciclo For

```
int voti[100];
int i = 0;

for (i=0; i<100; i++)
{
    scanf("%d", &voti[i]);
}
```

Stesso potere espressivo



# Esercizio 7

- Scrivere un programma che
  - legga due array di interi da tastiera
  - dica quale dei due array ha valor medio più alto
  - calcoli l'array concatenato tra i due array in ingresso
  - trovi tutti i numeri primi inseriti
  - trovi il massimo ed il minimo tra tutti i valori inseriti
  - dica in quale dei due array sono presenti i due valori di massimo e minimo
  - calcoli l'array riversato



# Esercizio 7: dichiarazione variabili

```
int i, j;
```

```
int a[DIM];
```

```
int b[DIM];
```

```
int ab[DIM2];
```

```
int ab_r[DIM2];
```

```
float medio_a, medio_b;
```

```
float somma = 0;
```

```
char primo = 1;
```



# Esercizio 7: lettura array

```
// LETTURA VALORI
// -----

// Leggo l'array A
for (i = 0; i < DIM; i++){
    printf("\nInserisci i valore %d per l'array a: ", i);
    scanf("%d",&a[i]);
}

// Stampo l'array A
printf("\narray a = ");
for (i = 0; i < DIM; i++)
    printf("%d ", a[i]);

// Leggo l'array B
for (i = 0; i < DIM; i++){
    printf("\nInserisci i valore %d per l'array b: ", i);
    scanf("%d",&b[i]);
}

// Stampo l'array B
printf("\narray b = ");
for (i = 0; i < DIM; i++)
    printf("%d ", b[i]);
```



# Esercizio 7: calcolo del valor medio

```
// VALOR MEDIO
// -----

// Calcolo valor medio si A

for (i = 0; i < DIM; i++){
    somma += a[i];
}
medio_a = somma / DIM;

// Calcolo valor medio si B
somma = 0;
for (i = 0; i < DIM; i++){
    somma += b[i];
}
medio_b = somma / DIM;

if(medio_a == medio_b)
    printf("L'array a e l'array b hanno lo stesso valor medio.\n");
else if (medio_a > medio_b)
    printf("L'array a ha valor medio piu' alto dell'array b.\n");
else if (medio_a < medio_b)
    printf("L'array b ha valor medio piu' alto dell'array a.\n");
```



# Esercizio 7: concatenazione

```
// CONCATENAZIONE
// -----

for (i=0; i < DIM; i++)
    ab[i] = a[i];

for (i=0; i < DIM; i++)
    ab[i+DIM] = b[i];

// Stampo l'array AB
printf("\narray a.b = ");
for (i = 0; i < DIM2; i++) {
    printf("%d ", ab[i]);
}

printf("\n");
```



# Esercizio 7: numeri primi

```
// NUMERI PRIMI
// -----

printf("I numeri primi inseriti sono: ");

for (i = 0; i < DIM2; i++){
    primo = 1; // Inizializzo la variabile
    for (j = 2; j < ab[i]; j++){
        if ((ab[i] % j) == 0 ){
            primo = 0;
        }

        if (primo == 1)
            printf("%d ", ab[i]);
    }

    printf("\n");
}
```



# Esercizio 7: massimo e minimo

```
// MAX e MIN
// -----

int max = 0;
int min = 2147483647;    //2^(n-1) - 1 :::: CPL2 max value
                        //n = 32 poiché int è rappresentato
                        //con 32 bit.

for (i = 0; i < DIM2; i++){

    if (ab[i] > max)
        max = ab[i];

    if (ab[i] < min)
        min = ab[i];

}

printf("MIN = %d\n", min);
printf("MAX = %d\n", max);
```



# Esercizio 7: massimo e minimo

```
// TROVARE DOVE SONO MAX e MIN
// -----

bool max_in_array_a = 0;
bool min_in_array_a = 0;

for (i = 0; i < DIM; i++){
    if (a[i] == max)
        max_in_array_a = 1;

    if(a[i] == min)
        min_in_array_a = 1;
}

printf("\n");

if (max_in_array_a)
    printf("Il valore massimo %d e' nell'array a", max);
else
    printf("Il valore massimo %d e' nell'array b", max);

printf("\n");

if (min_in_array_a)
    printf("Il valore minimo %d e' nell'array a", min);
else
    printf("Il valore minimo %d e' nell'array b", min);

printf("\n");
```



# Esercizio 7: array riversato

```
// ARRAY RIVERSATO
// -----

for (i = 0; i < DIM2; i++)
    ab_r[DIM2 - i - 1] = ab[i];

printf("\narray (a.b)_r = ");
for (i = 0; i < DIM2; i++){
    printf("%d ", ab_r[i]);
}

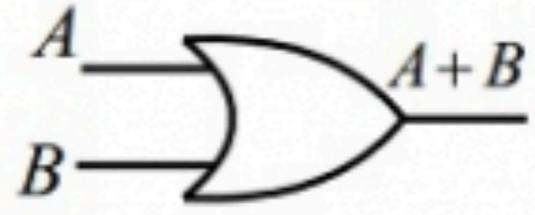
printf("\n");
printf("\n");
```

## AND



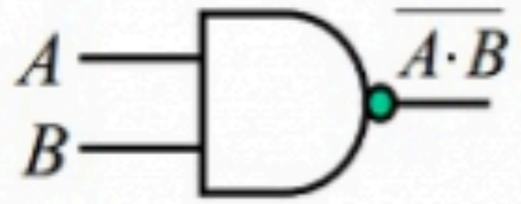
A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

## OR



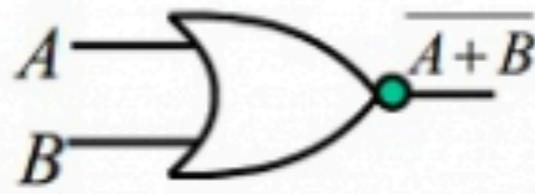
A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

## NAND



A	B	$\overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

## NOR



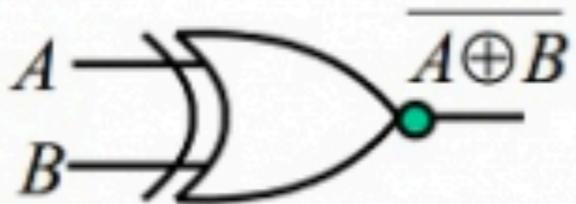
A	B	$\overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

## EXOR



A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

## EXNOR



A	B	$\overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1



# Note



# Note



# Note



# Note



# Note



# Note



# Note



# Note



# Note



# Note



# Note



# Note

**Tutte il materiale sarà  
disponibile sul mio sito  
internet!**

[www.alessandronacci.it](http://www.alessandronacci.it)

**See You Next Time!**

