

```
/*
* Politecnico di Milano
* Corso Informatica B - Prof. Loiacono
* Esercizi di: Alessandro A. Nacci
* Per contatti: alessandro.nacci@mail.polimi.it
* -----
* Esercitazione 3
* ESERCIZIO 1: COPIA DI UN ARRAY
* -----
* Scrivere un programma che inizializza un vettore monodimensionale
* di interi e poi copia il vettore in un altro vettore
* della stessa dimensione.
*/

#include <stdio.h>
main() {

    int i;
    int a[10]={1,2,3,4,5,6,7,8,9,10}; /* inizializzazione in fase di definizione */
    int b[10];

    /* stampa a[] */
    printf("a[]= ");

    for(i=0;i<10;i++){
        printf(" %d", a[i]);
    }

    /* copio a[] in b[] */
    printf("\ncopio a[] in b[]\n");
    for(i=0;i<10;i++)
        b[i]=a[i];

    /* stampa b[] */
    printf("b[]= ");

    for(i=0;i<10;i++){
        printf(" %d,", b[i]);
    }
}
```

```

/*
* Politecnico di Milano
* Corso Informatica B - Prof. Loiacono
* Esercizi di: Alessandro A. Nacci
* Per contatti: alessandro.nacci@mail.polimi.it
* -----
* Esercitazione 3
* ESERCIZIO 2: CRIVELLO DI ERATOSTENE
* -----
* Scrivere un programma che stampa i numeri primi minori di
* 100 (Crivello di Eratostene)
*/

/*  METODO DI ERATOSTENE
-----
Il procedimento  $\sqrt{n}$  il seguente: si scrivono tutti i naturali a partire
da 2 fino n in un elenco detto setaccio. Poi si cancellano (setacciano)
tutti i multipli del primo numero del setaccio (escluso lui stesso).
Si prosegue cos $\sqrt{\quad}$  fino ad arrivare in fondo. I numeri che restano sono i
numeri primi minori od uguali a n.    */

#include <stdio.h>

int main()
{
    int i,j;
    int array[100];

    // Inizializzazione dell'array
    for (i=0;i<100;i++)
        array[i]=1;

    // Si procede al setaccio...
    for (i=2;i<=100;i++) {
        if(array[i]==1) {
            for (j=2; j<(100/i); j++)
                array[i*j] = 0;
        }
    }

    // Si stampano i primi 100 numeri primi
    printf("\nNumeri primi < 100: ");
    for (i=1;i<100;i++){
        if(array[i]==1)
            printf("\t%d", i);
    }
}

```

```

/*
* Politecnico di Milano
* Corso Informatica B - Prof. Loiacono
* Esercizi di: Alessandro A. Nacci
* Per contatti: alessandro.nacci@mail.polimi.it
* -----
* Esercitazione 3
* ESERCIZIO 3: PRODOTTO SCALARE
* -----
* Scrivere un programma che calcoli il prodotto
* scalare tra due vettori.
*
* Esempio:
* [a b c d e] * [f g h i l]' = af + bg + ch + di + el
*
*/

```

```
#include <stdio.h>
```

```

int main()
{
    int i,j;
    int a[5]= {25, 35, 45, 55, 65};
    int b[5]= {23, 33, 43, 53, 63};

    int risultato = 0;

    // Stampo l'array A
    printf("\narray a = ");
    for (i = 0; i < 5; i++)
        printf("%d ", a[i]);

    // Stampo l'array B
    printf("\narray b = ");
    for (i = 0; i < 5; i++)
        printf("%d ", b[i]);

    printf("\nrisultato = ");
    for (i = 0; i < 5; i++){
        printf("%d * %d",a[i],b[i]);
        risultato += a[i] * b[i];

        if (i < 4)
            printf(" + ");
    }

    printf(" = %d\n", risultato);
}

```

```

/*
* Politecnico di Milano
* Corso Informatica B - Prof. Loiacono
* Esercizi di: Alessandro A. Nacci
* Per contatti: alessandro.nacci@mail.polimi.it
* -----
* Esercitazione 3
* ESERCIZIO 4: OPERAZIONI ARRAY
* -----
*
* Scrivere un programma che, dati letti 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
*
*/

#include <stdio.h>
#define DIM 5
#define DIM2 10

int main()
{
    typedef enum{false, true} bool;           //false = 0; true = 1 --> importante questo
    ordine                                   //per poter fare gli 'if(true)''

    int i, j;
    int a[DIM];
    int b[DIM];

    int ab[DIM2];
    int ab_r[DIM2];

    float medio_a, medio_b;

    // LETTURA VALORI
    // -----

    // Leggo l'arrayA
    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]);

// VALOR MEDIO
// -----

// Calcolo valor medio si A
float somma = 0;
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");

// 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");

// NUMERI PRIMI
// -----

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

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

    bool primo = true;    //booleana
    for (j = 2; j < ab[i]; j++)
        if ((ab[i] % j) == 0 ){
            primo = false;
            break;
        }

    if (primo)

```

```

        printf("%d ", ab[i]);
    }

printf("\n");

// 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);

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

bool max_in_array_a = false;
bool min_in_array_a = false;

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

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

printf("\n");

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

printf("\n");

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

printf("\n");

// 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");
}
```