

Cognome e Nome		Corso		Matricola	
----------------	--	-------	--	-----------	--

Traccia A

Esercizio 1

Si consideri il seguente codice:

```
static int valuta( int vet[] ) {
    int p = 0, lu = vet.length;
    int j = lu/2;

    if(lu%2 !=0)
        return -1 ;
    while( j < lu) {
        if ( vet[j] % vet[j-(lu/2)] == 0 || vet[j-(lu/2)] % vet[j] == 0 )
            p += vet[j];
        j++;
    }
    return p;
}
```

Si descriva sinteticamente il significato della funzione svolta dal metodo. Inoltre si mostri la traccia d'esecuzione ed il valore restituito dal metodo nel caso in cui il parametro sia **vet** = [10, 2, 15, 5, 3, 5].

Esercizio 2

Si realizzi un metodo *elabora* che, ricevuti in ingresso due vettori **v1** e **v2** di interi restituisce **false** se i due vettori non hanno la stessa dimensione. Se entrambi hanno la stessa dimensione, il metodo restituisce **true** se la somma degli elementi di **posizione dispari** di **v1** è uguale al prodotto degli elementi con **valore pari** di **v2**. Ad esempio, se **v1** = [1, 2, 0, 16, 4, -2] e **v2** = [1 , 2 , 4 , 7, 2, 1], il valore restituito è **true** perchè **v1[1] + v1[3] +v1[5] = v2[1] * v2[2] * v2[4] = 16**.

Esercizio 3

Si vuole progettare una classe *Matrix* che permetta di trasformare una matrice quadrata M, le cui componenti (quadrate) sono mostrate in figura:

$$M = \begin{matrix} A & C \\ B & D \end{matrix}$$

In una nuova matrice

$$M' = \begin{matrix} E & O \\ O & F \end{matrix}$$

dove **E** è una matrice contenente elementi nulli, eccetto gli elementi sulla diagonale (che corrispondono agli elementi della diagonale di A*B). Analogamente, **F** è una matrice contenente elementi nulli eccetto gli elementi sulla diagonale (che corrispondono agli elementi della diagonale di C*D).

Si realizzino a questo proposito i seguenti metodi:

- 1) un metodo *estrai* che riceve in ingresso una matrice *M* di interi e un intero *i* che può assumere valori 1, 2, 3, 4 e restituisce rispettivamente le sottomatrici quadrate A, B, C, D.
- 2) Un metodo *prodotto*, che riceve due matrici quadrate e ne restituisce il prodotto;
- 3) Un metodo *estraiDiagonale*, che riceve una matrice X e restituisce una matrice Y tale che Y[i][i] = X[i][i], e Y[i][j] = 0 con i ≠ j;
- 4) Un metodo *main* che legge da input quattro matrici quadrate A, B, C, D e restituisce la matrice ottenuta M' componendo le sottomatrici come in figura.