

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

Traccia C

Esercizio 1

Si consideri il seguente codice:

```
static int metodoesempio(int[] x)
{
    int m=0;
    if (x.length <=1)
        return -1;
    else
    {
        int n;
        if (x.length%2 == 1)
            n = x.lenght -1;
        else
            n = x.length;
        for (int i=0; i < n/2; i++)
            if ((x[i]+x[n-1-i]) > m)
                m = x[i]+x[n-1-i];
        return m;
    }
}
```

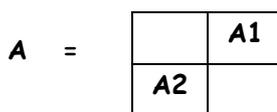
Si assuma che i valori contenuti in x siano positivi e si descriva sinteticamente la funzione svolta dal metodo e, in particolare, se ne mostri l'esecuzione nel caso in cui il valore dei parametri sia $x=[10,12,6,21,7]$. Specificare il contenuto dell'intero **m** restituito nel caso indicato.

Esercizio 2

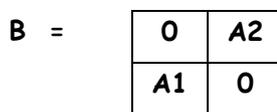
Scrivere un metodo *componivett* che riceve due array **V1** e **V2** di **n** interi e ritorna un vettore **V3** di uguale lunghezza **n** in cui l'elemento di posizione **i** è pari alla somma di tutti gli elementi di **V2** che hanno valore maggiore del valore di **V1[i]**. Ad esempio, se **V1** = [3, 5, 11, 10], **V2** = [1, 4, 14, 6] allora il risultato sarà **V3** = [24,20, 14,14].

Esercizio 3

Si deve realizzare un programma Java per la gestione di matrici quadrate di numeri interi. Il programma deve in particolare (a) estrarre da una matrice quadrata **A** di dimensione **n** (pari), due sottomatrici (**A1** e **A2**) ognuna di dimensione **n/2**:



(b) calcolare la matrice



dove le sottomatrici indicate da **O** contengono tutti elementi nulli.

e, infine,

c) verificare se esiste una colonna di **A2** che sia uguale ad una colonna di **A1**.

Strutturare il programma (almeno) nei seguenti metodi:

- *estrai* – che riceve una matrice quadrata **A** e un numero intero **i** (che può assumere i valori 1 e 2) e restituisce la sottomatrice **A_i**, cioè una delle sottomatrici **A1**, **A2**;
- *componi* - che date due matrici **A1** e **A2** quadrate compone una matrice **B** come in figura.
- *controllacolonneuguali* – che riceve una coppia di matrici della stessa dimensione e verifica (restituendo un valore booleano) se almeno una colonna della prima matrice è uguale ad una colonna della seconda matrice;
- *main* - che legge una matrice **A** di numeri reali e provvede, invocando opportunamente i metodi proposti, a realizzare i compiti di cui ai punti (a), (b) e (c), visualizzando su output la matrice **B** e l'esito della verifica (c).