

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

Traccia B**Esercizio 1**

Si consideri il seguente metodo

```
public static boolean esercizio(char[] vett, int d){
    int s = 0;
    while ((s < d-1) && (vett[s] == vett[d-1]))
    {
        s++;
        d--;
    }
    if (s < d-1)
        return false;
    return true;
}
```

Si descriva sinteticamente l'algorithmo che esso implementa e, in particolare, se ne mostri l'esecuzione nel caso in cui l'input sia $v = [a, b, c, s, 6, b, a]$ e $d = 7$.

Esercizio 2

Scrivere un metodo *sostituisci* che legga una **stringa di caratteri** con valori '0' o '1' e se il primo elemento della sequenza è pari a '1', ogni volta che trova due '1' consecutivi li sostituisce con un '0' e ritorna la stringa risultante.

Esercizio 3

Si scriva una classe *GestMatrici* per l'elaborazione di matrici (rettangolari o quadrate) di numeri interi di dimensione massima 10x10. Tale classe deve includere (almeno) i seguenti metodi:

- *lettura*: riceve una matrice di interi **A** e ne legge il contenuto da tastiera per righe, partendo dall'ultima riga sino alla prima. Il metodo restituisce la matrice letta.
- *scrittura*: riceve una matrice di interi **A** e ne scrive il contenuto su output.
- *ePalindroma*: riceve una matrice di interi **A** e restituisce *true* se il contenuto di **A** è palindromo, false altrimenti. Una matrice è da intendersi palindroma quando la prima riga è uguale all'ultima, la seconda alla penultima ecc.
- *prodottoScalare*: riceve una matrice di interi **A** e due indici di riga r_1 e r_2 e calcola e ritorna il prodotto scalare tra la riga r_1 e la riga r_2 .
- *main*: legge da tastiera le dimensioni di una matrice di interi m . Quindi provvede a inizializzarne il contenuto con la lettura da input. Successivamente il programma visualizza la matrice, scrive se essa è palindroma o meno e, infine, stampa il prodotto scalare tra prima e l'ultima riga della matrice stessa.

Esempio: la seguente matrice 5x5 è palindroma.

1	2	3	4	5
-5	10	20	30	-2
9	8	7	6	0
-5	10	20	30	-2
1	2	3	4	5

Esercizio 4

Scrivere un metodo *distinti* che riceva un array di interi **v** e restituisca un nuovo array contenente tutti e soli gli elementi distinti di **v**. Ad esempio, se il contenuto di **v** è [2,1,5,2,6,2,5,9,3,1] allora l'array da restituire è [1,2,3,5,6,9].