

## – Prima Prova Intermedia – 30/01/2023

---

Questa prova intermedia è costituita da due esercizi. Potranno essere consegnati al docente solo gli esercizi che avranno superato i nostri JUnit Test. I test svolti dagli studenti non saranno presi in considerazione per la valutazione. La consegna di almeno un esercizio è condizione necessaria, ma non sufficiente, per passare la prova.

Per la valutazione saranno presi in considerazione aspetti di “buona programmazione”, “pulizia del codice” ed efficienza. Ad es.: formattazione corretta del codice, rendere il codice modulare aggiungendo ove necessario altri metodi rispetto a quelli richiesti dall’esercizio, soprattutto se questi rendono il codice più pulito e leggibile, o se evitano duplicazione di codice. Inoltre, non ci devono essere warning nel codice scritto.

IMPORTANTE: seguire attentamente le specifiche per quanto riguarda i nomi dei metodi e la firma dei metodi, altrimenti i test automatici falliranno rendendo il compito insufficiente.

### CONSEGNA ESERCIZI:

Entro il termine ultimo previsto per la consegna dello scritto, gli studenti che intendono consegnare provvedono a caricare il sorgente (o i sorgenti) **.javamm** attraverso l’apposita attività “compito”, disponibile nella pagina del corso al seguente link:


<https://e-val.unifi.it/mod/assign/view.php?id=25314>

Fino allo scadere del tempo, lo studente potrà apportare modifiche al proprio lavoro. Non saranno accettate consegne effettuate in ritardo, o con modalità diverse da quelle definite dal docente. All’orario stabilito ad inizio compito il docente dichiara finita la prova e chiude la sessione.

## Esercizio Java-- n. 1: Rimozione cifre da intero

Sia  $n$  un numero intero (di tipo `long`) con  $n \geq 0$ , e sia  $c$  un numero intero (di tipo `short`) con  $1 \leq c \leq 9$ . La rimozione della cifra  $c$  dal numero  $n$  è una procedura che, partendo dalla cifra meno significativa di  $n$  e procedendo verso le cifre più significative, rimuove da  $n$  tutte le sequenze (non sovrapposte) composte da  $c$  cifre consecutive uguali alla cifra  $c$ .

Ad esempio, sia  $n=32221222$  e  $c=2$ , la rimozione della cifra 2 dal numero  $n$  genera il numero intero  $r=321$  poiché, leggendo le cifre da destra (cifre meno significative) verso sinistra (cifre più significative), si incontrano e si rimuovono le seguenti sequenze composte dal numero 2 ripetuto 2 volte:


 $n = 3 \ 2 \ \boxed{2 \ 2} \ 1 \ \boxed{2 \ 2} \ \boxed{2 \ 2} \rightarrow r = 3 \ 2 \ 1$ 
(es. 1)

Altri esempi:

- Con  $n = 3 \boxed{1} \ 4 \ 2 \ 2 \ \boxed{1} \ \boxed{1}$  e  $c=1$ , la procedura genera l'intero  $r=3422$ ; (es. 2)
- Con  $n = 3 \ 3 \ \boxed{3 \ 3 \ 3} \ 0 \ 0$  e  $c=3$ , la procedura genera l'intero  $r=3300$ ;
- Con  $n = 1 \ \boxed{3 \ 3 \ 3} \ \boxed{3 \ 3 \ 3}$  e  $c=3$ , la procedura genera l'intero  $r=1$ ;
- Con  $n = \boxed{4 \ 4 \ 4 \ 4}$  e  $c=4$ , la procedura genera l'intero  $r=0$ ;
- Con  $n = \boxed{3 \ 3 \ 3} \ 0 \ 0 \ 0$  e  $c=3$ , la procedura genera l'intero  $r=0$ ;
- Con  $n = 3 \ 3 \ 1 \ 2 \ 2 \ 3 \ 1$  e  $c=3$ , la procedura genera l'intero  $r=3312231$  (non ci sono sequenze composte da 3 cifre consecutive uguali a 3).

**[DIFFICOLTA' STANDARD]:** Scrivere un metodo Java--, chiamato **rimuoviCifre**, che dati in input un numero intero  $n$  (di tipo `long`) con  $n \geq 0$ , e un numero intero  $c$  (di tipo `short`) con  $1 \leq c \leq 9$ , restituisca il numero intero (di tipo `long`) generato dalla rimozione della cifra  $c$  dal numero  $n$  come precedentemente descritto.

**[DIFFICOLTA' RIDOTTA]:** Si può assumere che  $c$  possa essere uguale soltanto a 1 o a 2 (vedere es. 1 ed es. 2).

## NOTA BENE:

- Saranno premiate le soluzioni che dichiareranno meno variabili di appoggio e occuperanno quindi meno memoria dati.
- E' possibile invocare il metodo `Math.pow` per calcolare la potenza di un numero.
- Nello svolgere l'esercizio NON devono essere utilizzati i metodi `clone`, o `arraycopy`, o metodi della classe `Arrays`. L'utilizzo di tali metodi renderà l'esercizio automaticamente insufficiente.
- Gli studenti dovranno consegnare per questo esercizio solo 1 sorgente relativo alla soluzione con "difficoltà ridotta" oppure relativo alla soluzione con "difficoltà standard".

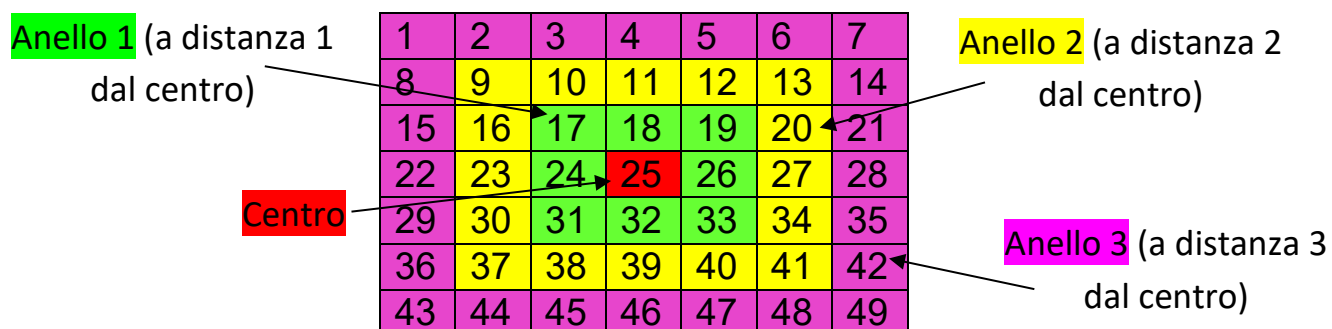
## JUnit Test:

- I JUnit Test da superare per la "difficoltà ridotta" sono quelli della classe **RimuoviCifreRidottaTest** (i test della classe `RimuoviCifreStandardTest` falliranno).
- I JUnit Test da superare nel caso di "difficoltà completa" sono solo quelli della classe **RimuoviCifreStandardTest**.

## Esercizio Java-- n. 2: Rotazioni Anelli di Matrice

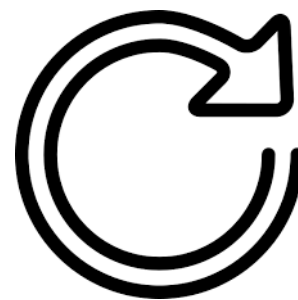
Sia  $T$  una matrice quadrata di interi di dimensione  $m \times m$ , con  $m \geq 3$  e  $m$  dispari. Sia  $k$  un intero, con  $0 < k \leq m/2$ .

Informalmente, si definisce anello  $k$ -mo della matrice  $T$  come l'insieme di celle di  $T$  che “distano”  $k$  celle dal centro della matrice. Ad esempio, in una matrice  $T$  di dimensione  $7 \times 7$  si possono definire tre anelli a distanza 1, 2 e 3 dal centro, come nella seguente figura (esempio 1):



Una rotazione dell'anello  $k$  della matrice  $T$  modifica la matrice ruotando in senso orario di 1 posizione tutti gli elementi dell'anello  $k$  della matrice. Ad esempio, la rotazione dell'anello 2 della matrice dell'esempio precedente modifica la matrice in questo modo:

1	2	3	4	5	6	7
8	16	9	10	11	12	14
15	23	17	18	19	13	21
22	30	24	25	26	20	28
29	37	31	32	33	27	35
36	38	39	40	41	34	42
43	44	45	46	47	48	49



**[CONSEGNA OBBLIGATORIA]:** Scrivere un metodo Java--, chiamato **ruotaAnelloMatrice**, che dati in input una matrice quadrata  $T$  di interi di dimensione  $m \times m$  (con  $m \geq 3$  e  $m$  dispari), ed un intero  $k$  (con  $0 < k \leq m/2$ ), restituisca la matrice  $T$  modificata dalla rotazione del suo anello  $k$  come precedentemente descritto.

**[CONSEGNA EXTRA - FACOLTATIVA]:** Scrivere un metodo Java--, chiamato **ruotaAnelloMatriceNposizioni**, che dati in input una matrice quadrata  $T$  di interi di dimensione  $m \times m$  con  $m \geq 3$  e  $m$  dispari, un intero  $k$  con  $0 < k \leq m/2$ , ed un numero intero  $n$ , restituisca la matrice  $T$  modificata dalla rotazione del suo anello  $k$  di  $n$  posizioni in senso orario (se  $n > 0$ ) o in senso anti-orario (se  $n < 0$ ). Ad esempio, la rotazione dell'anello  $k=2$  della matrice  $T$  di partenza (vedi esempio 1) di 3 posizioni in senso orario (quindi con  $n=3$ ) modifica la matrice in questo modo:

1	2	3	4	5	6	7
8	30	23	16	9	10	14
15	37	17	18	19	11	21
22	38	24	25	26	12	28
29	39	31	32	33	13	35
36	40	41	34	27	20	42
43	44	45	46	47	48	49

**NOTA BENE:**

- I Junit Test da superare per la consegna obbligatoria sono quelli della classe **ruotaAnelloMatriceTest** (quelli della classe `ruotaAnelloMatriceNposizioniTest` falliranno).
- I Junit Test da superare per la consegna extra (facoltativa) sono quelli della classe **ruotaAnelloMatriceNposizioniTest**.
- Nello svolgere l'esercizio NON devono essere utilizzati i metodi `clone`, o `arraycopy`, o metodi della classe `Arrays`. L'utilizzo di tali metodi renderà l'esercizio automaticamente insufficiente.