

Scrivere un programma C++ che legga da input nell'ordine un intero **N** e una matrice quadrata **A** di dimensione **N** contenente caratteri.

Ogni carattere alfabetico minuscolo corrisponde ad un numero intero, ovvero **a** corrisponde ad **1**, **b** a **2**, **c** a **3** e così via fino a **z**. I restanti caratteri corrispondono a **0**.

Considerando i quattro vertici della matrice, si calcola la somma dei numeri corrispondenti a ciascuno dei quattro caratteri. A questo punto si calcola il resto della divisione tra la somma e 26 e vi si aggiunge 1. Se tale numero corrisponde ad un carattere presente nella matrice **A**, allora tale carattere viene memorizzato.

A questo punto si considera la matrice inscatolata nella precedente, ovvero quella che si ottiene dalla precedente escludendo la cornice. Si considerano i quattro vertici di questa nuova matrice. Si calcola nuovamente la somma dei numeri corrispondenti a ciascuno dei quattro caratteri. Se il risultato della divisione tra la somma e 26 a cui si aggiunge 1, corrisponde ad un carattere presente nella matrice **A** originaria, allora tale carattere viene memorizzato.

Si reitera questo procedimento fino a quando possibile, ovvero fino a quando la matrice inscatolata ha dimensione maggiore o uguale a 2.

Si stampino in output i caratteri memorizzati.

ATTENZIONE: Stampare solo i caratteri senza aggiungere altre spaziature o endl.

ESEMPIO: Sia **N=8** e sia **A** la seguente matrice di caratteri:

a	4	g	L	m	g	p	t
S	m	H	y	P	f	T	z
4	z	k	w	e	y	m	t
g	6	g	a	r	p	g	h
g	w	7	a	r	s	L	y
3	z	w	*	r	z	R	y
S	2	f	b	b	b	&	z
N	e	d	K	z	r	y	i

(1) I caratteri presenti nei quattro vertici della matrice **A** sono **a**, **t**, **N**, **i**, corrispondenti ai numeri **1**, **20**, **0**, **9**. La cui somma è **30**. Il resto della divisione di 30 per 26 è 4 e aggiungendovi 1 come richiesto, otteniamo **5**, che corrisponde al carattere minuscolo **e**. Poiché il carattere **e** è presente nella matrice **A** (una sua occorrenza è in posizione [2][4]), tale carattere viene memorizzato.

(2) Si considera poi la matrice inscatolata di dimensione 6 i cui quattro vertici sono **m**, **T**, **2**, **&**, corrispondenti ai numeri **13**, **0**, **0**, **0**. La cui somma è **13**. Il resto della divisione di 13 per 26 è 13 e aggiungendovi 1 come richiesto, otteniamo **14**, che corrisponde al carattere minuscolo **n**. Poiché il carattere **n** non è presente nella matrice **A**, tale carattere non viene memorizzato.

(3) Si considera poi la matrice inscatolata di dimensione 4 i cui quattro vertici sono **k**, **y**, **w**, **z**, corrispondenti ai numeri **11**, **25**, **23**, **26**. La cui somma è **85**. Il resto della divisione di 85 per 26 è 7 e aggiungendovi 1 come richiesto, otteniamo **8**, che corrisponde al carattere minuscolo **h**. Poiché il carattere **h** è presente nella matrice **A** (una sua occorrenza è in posizione [3][7]), tale carattere viene memorizzato.

(4) Infine si considera la matrice inscatolata di dimensione 2 i cui quattro vertici sono **a**, **r**, **a**, **r**, corrispondenti ai numeri **1**, **18**, **1**, **18**. La cui somma è **38**. Il resto della divisione di 38 per 26 è 12 e aggiungendovi 1 come richiesto, otteniamo **13**, che corrisponde al carattere minuscolo **m**. Poiché il carattere **m** è presente nella matrice **A** (una sua occorrenza è in posizione [0][4]), tale carattere viene memorizzato.

In output si stamperà: **ehm**