

**Traccia A**

Cognome		Nome		Matricola	
---------	--	------	--	-----------	--

**Esercizio 1 (Programmazione in C, 8 punti)**

Scrivere un programma in C per la gestione di matrici di float, che contenga i seguenti metodi:

Una funzione create, che riceve due interi (dimensione di righe e colonne), crea una matrice delle dimensioni indicate, e restituisce il suo puntatore.

Un metodo read per la lettura di una matrice da tastiera.

Un metodo print per la stampa della matrice su output.

Un metodo **verifica\_vet\_matrice**, a cui viene passato fra gli altri parametri anche un vettore V, e che restituisce (in maniera opportuna usando i puntatori) 2 interi, **matSup** e **vetInf**:

il primo (**matSup**) corrisponde al numero di righe pari che sono più grandi della corrispondente riga dispari (la 2 rispetto alla 1, la 4 rispetto alla 3. E così via). Per più grande si intende che almeno la metà più uno degli elementi siano maggiori degli elementi corrispondenti dell'altra riga.

il secondo (**vetInf**) è il numero di colonne della matrice in cui almeno metà +1 elementi siano più grandi del corrispondente elemento del vettore V (si suppone che la dimensione di V è pari al numero di righe della matrice).

Il metodo main che dichiara e alloca la matrice e invoca opportunamente i metodi precedenti.

NOTA: QUALSIASI SOLUZIONE ALTERNATIVA CHE NON USI I PUNTATORI SARA' CONSIDERATA ERRATA.

**Esercizio 2 (Programmazione Shell, 10 punti)**

Scrivere un programma shell che riceva da linea di comando tre argomenti: i primi 2 sono due cartelle CartA e CartB e il terzo è un file elezioni.txt.

Il programma dovrà leggere dal file elezioni.txt i dati, ripetuti per ogni riga: nome\_candidato e votiCandidato.txt. (Esempi: Rossi VotiRossiPrimasezione.txt). Lo stesso candidato può essere ripetuto più volte.

Per ogni riga, il programma dovrà verificare se esiste nella cartella CartA un file nome\_candidato.doc e il file votiCandidato.txt. Nel caso esistano i 2 file dovrà inserire nella cartella CartB un file nomecandidato.txt (Esempio Rossi.txt) in cui inserire i voti del candidato (appendendoli in coda).

Alla fine, per ogni candidato, dovranno essere stampati i voti totali.

Il programma deve gestire i casi d'eccezione (numero di argomenti diverso da tre, file F già esistente, ecc.) interrompendo l'esecuzione con un messaggio all'utente.

**Esercizio 3 (Programmazione System Call di Linux, 12 punti)**

Realizzare un programma C che, utilizzando le System Call di Linux, svolga le seguenti funzioni.

Un padre creerà un processo Figlio F1 e F2, mentre il solo F1 creerà un altro processo N (nipote). Si utilizzino le pipe che si ritengono necessarie per far comunicare i vari processi. Il padre leggerà un vettore di n interi positivi da un file binario, quindi leggerà un intero X da tastiera. A questo punto passerà al figlio F1 tutti i numeri pari, mentre a F2 i numeri dispari. Il processo F1 troverà il massimo dei numeri pari e passerà al nipote tutti i numeri, mentre F2 troverà il minimo e lo restituirà al padre. Il nipote troverà la media di tutti i numeri minori di X e scriverà tale media su un file di testo "Output.txt".