

**UNIVERSITA' di VERONA**  
**DIPARTIMENTO di INFORMATICA**  
**CORSO DI LAUREA in BIOINFORMATICA**  
Elementi di architettura e sistemi operativi  
2017/2018

**CONSEGNA - ELEARNING: entro 23.00 del 15 Maggio 2018**

**Elaborato: 1 - SHELL**

Utilizzo della shell e sua programmazione.

Scrivere uno script bash "*gestioneAnagrafica.sh*" per la gestione anagrafica di studenti universitari.

Per **ogni entry dell'anagrafica** i dati vengono salvati su una riga di un file txt, separando ogni dato con un punto e virgola ";" (formato CSV). I dati saranno organizzati secondo il seguente schema:

<nome>;<cognome>;<matricola>;<anno>

Si assuma che la matricola identifichi in modo univoco ogni studente, ossia che nessuno studente inserito in anagrafe possieda la stessa matricola.

I dati da salvare sono:

1. Nome
2. Cognome
3. Matricola (univoco composto da 6 numeri, es. 000123 o 012345)
4. Anno (anno di corso, valori ammessi 1,2,3)

Eseguendo lo script, apparirà il seguente menu interattivo:

1. Aggiungi
2. Elimina
3. Visualizza elenco
4. Ricerca
5. Iscritti per anno
6. Esci

Il comportamento per ciascuna voce del menu deve essere il seguente.

**1. AGGIUNGI:**

Il comando *Aggiungi* consente di aggiungere un nuovo studente all'anagrafe. Verrà chiesto all'utente di inserire in ordine nome, cognome, matricola e anno, leggendo l'input da tastiera dell'intera riga per ogni elemento inserito, come ad esempio:

```
Inserire nome:
Pippo
Inserire cognome:
Pluto
Inserire matricola:
123456
Inserire anno:
2
```

Al termine dell'inserimento, le informazioni relative al nuovo studente verranno aggiunte in coda al file contenente i dati secondo il formato specificato in precedenza.

**2. ELIMINA:**

Il comando *Elimina* consente di eliminare uno studente dall'anagrafe. Verrà richiesto di inserire la matricola dello studente che si vuole eliminare

# UNIVERSITA' di VERONA

## DIPARTIMENTO di INFORMATICA

dall'anagrafe. Se la matricola fornita non corrisponde ad alcun studente in anagrafe, verrà visualizzato un messaggio di errore. Se invece la matricola corrisponde ad uno studente in anagrafe, la riga contenente le informazioni relative a tale studente verrà eliminata dal file.

### 3. VISUALIZZA ELENCO

Il comando *Visualizza elenco* consente di visualizzare in formato tabulare l'elenco degli studenti in anagrafe. L'elenco deve essere ordinato per matricola crescente. Un possibile output fornito dal comando potrebbe essere il seguente:

```
NOME COGNOME MATRICOLA ANNO
Tizio Bianchi 111111 2
Caio Rossi 2222222 3
```

I vari campi relativi ad ogni studente devono essere separati da almeno uno spazio (o una tabulazione o piu' spazi). **Non è richiesto che i vari campi degli studenti siano allineati alle rispettive colonne.**

### 4. RICERCA

Il comando *Ricerca* consente di ricercare uno o più studenti per cognome. Verrà richiesto di inserire il cognome che si intende ricercare. Verranno quindi mostrate le informazioni di ogni studente in anagrafe avente il cognome cercato stampando una riga per ogni campo. **Gli studenti mostrati dovranno essere ordinati per nome.** Ad esempio, cercando il cognome "Bianchi", l'output dovrebbe essere il seguente:

```
Nome: Tizio
Cognome: Bianchi
Matricola: 111111
Anno: 2
```

Se non esistono studenti in anagrafe con il cognome cercato, lo script dovrà visualizzare un messaggio che specifica l'assenza di studenti aventi tale cognome.

### 5. ISCRITTI

Il comando *Iscritti per anno* dovrà visualizzare il numero totale di studenti iscritti a ciascun anno di corso. Si assuma che il campo anno possa assumere solo i valori interi 1, 2 o 3. L'output del comando potrebbe essere il seguente:

```
Primo anno: 3
Secondo anno: 5
Terzo anno: 1
```

### 6. ESCI

Il comando *esci* termina l'esecuzione del programma.

Lo script ***gestioneAnagrafica.sh*** potrà prendere in input il nome/percorso del database dell'anagrafica (***gestioneAnagrafica.sh /tmp/studenti.txt***) o qualora non venga specificato utilizzerà il nome/percorso implementato per default nello script.

Il programma deve girare senza errori...

È titolo preferenziale per la valutazione (ma non obbligatorio) il controllo dei casi particolari, come ad esempio digitare **9** nella scelta del menu iniziale (ovvero l'inserimento di input o file non previsti).

L'utilizzo di ricorsione non necessaria comporta una penalizzazione (es. richiamare il menu al posto di uscire dalla funzione).

La mancanza di alcuni punti richiesti comporta una "penalizzazione" sulla valutazione ma può portare a superare comunque l'esame.

**UNIVERSITA' di VERONA**  
**DIPARTIMENTO di INFORMATICA**

L'elaborato è personale, la presentazione di elaborati identici comporta la NULLITÀ degli elaborati UGUALI.

In calce al file come commento deve essere riportato il seguente commento:

#Matricola  
#Nome e cognome  
#Data di realizzazione  
#Titolo esercizio

Riferimenti:

Nicola Drago: [nicola.drago@univr.it](mailto:nicola.drago@univr.it)

Marco Panato: [marco.panato@studenti.univr.it](mailto:marco.panato@studenti.univr.it)

CONSEGNA TRAMITE E-LEARNING

ENTRO 23:00 del 15 Maggio 2018

# UNIVERSITA' di VERONA

## DIPARTIMENTO di INFORMATICA

### **FAQ**

*1. E' possibile inserire il codice su script separati, oppure bisogna usare le funzioni?*

Questa decisione è lasciata allo studente, che può scegliere il modo più opportuno. È quindi possibile sviluppare il programma inserendo il codice su funzioni oppure su file separati.

*2. È possibile usare il comando comm/sed/awk/etc.?*

Sì. Qualsiasi comando bash è consentito, anche se non spiegato a lezione. Lo studente è però tenuto a conoscere i comandi che usa nel codice. Non è lecito usare script in altri linguaggi, quali ad esempio Perl o TCL.

N.B.: Tutto quanto non esplicitato in questo documento può essere implementato liberamente.