

GRUPPO 2 - Pseudocodice funzioni Cancellazione

FUNZIONE “CancellareUtente”

INPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi
----------	---------------	---

idUtenteDaCancellare	tipo intero	valore indicante l'id della struttura di tipo Utente da cancellare
----------------------	-------------	--

OUTPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema aggiornate
----------	---------------	---

DATI DI LAVORO

vettoreUtente	tipo vettore Utente	vettore degli utenti del database
---------------	---------------------	-----------------------------------

vettoreIdPlaylist	tipo vettore di interi	vettore degli id playlist dell'utente da cancellare
-------------------	------------------------	---

i	tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$	valore indicante la posizione nel vettoreUtente di UtenteDaEliminare
---	--	--

j	tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$	valore indicante la posizione nel vettoreIdPlaylist da considerare
---	--	--

PSEUDOCODICE

vettoreUtente \leftarrow LeggereVettoreUtente(BaseDati)

i \leftarrow 0

Mentre(LeggereIdUtente(LeggereUtente(vettoreUtente,i)) \neq idUtenteDaCancellare) AND (i $<$ ELEM_MAX)

i \leftarrow i+1

FINE

Se(i = ELEM_MAX)

Allora stampa(“Utente selezionato non esistente”)

Altrimenti

vettoreUtente \leftarrow SpostareSinistraVettoreUtente(vettoreUtente,i)

BaseDati \leftarrow ScrivereVettoreUtente(vettoreUtente, BaseDati)

vettoreIdPlaylist \leftarrow InizializzareVettore(vettoreIdPlaylist)

vettoreIdPlaylist \leftarrow RicercarePlaylistIdUtente(idUtenteDaCancellare, BaseDati)

j \leftarrow 0

Mentre(jesimo elemento di vettoreIdPlaylist \neq SENTINELLA)

BaseDati \leftarrow CancellarePlaylist(BaseDati, jesimo elemento di vettoreIdPlaylist)

j \leftarrow j+1

FINE

FINE

FUNZIONE “CancellarePlaylist”

INPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi
----------	---------------	---

idPlaylistDaCancellare	tipo intero	valore indicante l'id della struttura di tipo Playlist da cancellare
------------------------	-------------	--

OUTPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema aggiornate
----------	---------------	---

DATI DI LAVORO

vettorePlaylist	tipo vettore Playlist	vettore delle playlist del database
-----------------	-----------------------	-------------------------------------

vettoreBranoPlaylist	tipo vettore BranoPlaylist	vettore dei collegamenti BranoPlaylist del database
----------------------	----------------------------	---

i	tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$	valore indicante la posizione nel vettorePlaylist di PlaylistDaEliminare
---	--	--

posizioneCollegamento	tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$	valore indicante la posizione nella tabella di un collegamento brano-playlist
-----------------------	--	---

PSEUDOCODICE

vettorePlaylist ← LeggereVettorePlaylist(BaseDati)

i ← 0

Mentre(LeggereIdPlaylist(LeggerePlaylist(vettorePlaylist,i)) <> idPlaylistDaCancellare)AND(i < ELEM_MAX)

i ← i+1

FINE

Se(i = ELEM_MAX)

Allora stampa("Playlist selezionata non esistente")

Altrimenti

vettorePlaylist ← SpostareSinistraVettorePlaylist(vettorePlaylist, i)

BaseDati ← ScrivereVettorePlaylist(vettorePlaylist, BaseDati)

vettoreBranPlaylist ← LeggereVettoreBranPlaylist(BaseDati)

Mentre(posizioneCollegamento <> SENTINELLA)

posizioneCollegamento ← RicercaPosizioneBranPlaylistIdPlaylist(idPlaylistDaCancellare, vettoreBranPlaylist)

Se(posizioneCollegamento <= ELEM_MAX)AND(posizioneCollegamento <> SENTINELLA)

Allora

vettoreBranPlaylist ← SpostareSinistraVettoreBranPlaylist(vettoreBranPlaylist, posizioneCollegamento)

BaseDati ← ScrivereVettoreBranPlaylist(vettoreBranPlaylist, BaseDati)

Altrimenti

stampa("Playlist non collegata a nessun brano")

FINE

FINE

FINE

FUNZIONE "CancellareGenere"

INPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi
----------	---------------	---

idGenereDaCancellare	tipo intero	valore indicante l'id della struttura di tipo Genere da cancellare
----------------------	-------------	--

OUTPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema aggiornate
----------	---------------	---

DATI DI LAVORO

esito	tipo booleano	"vero" se ho correttamente eliminato il genere e i suoi collegamenti ad altre strutture, "falso" altrimenti
-------	---------------	---

vettoreGenere	tipo vettore Genere	vettore dei generi del database
---------------	---------------------	---------------------------------

vettoreBranGenere	tipo vettore BranoGenere	vettore dei collegamenti BranoGenere del database
-------------------	--------------------------	---

i	tipo intero >= 0 e <= ELEM_MAX	valore indicante la posizione nel vettoreGenere di GenereDaEliminare
---	--------------------------------	--

j	tipo intero >= 0 e <= ELEM_MAX	valore indicante la j-esima iterazione alla ricerca di un collegamento brano-genere nel vettoreBranGenere
---	--------------------------------	---

posizioneCollegamento	tipo intero >= 0 e <= ELEM_MAX	valore indicante la posizione nella tabella di un collegamento brano-genere
-----------------------	--------------------------------	---

PSEUDOCODICE

vettoreGenere ← LeggereVettoreGenere(BaseDati)

i ← 0

Mentre(LeggereIdGenere(LeggereGenere(vettoreGenere,i)) <> idGenereDaCancellare)AND(i < ELEM_MAX){

i ← i+1

FINE

Se(i = ELEM_MAX)

Allora stampa("Genere selezionato non esistente")

Altrimenti

```

vettoreGenere ← SpostareSinistraVettoreGenere(vettoreGenere, i)
BaseDati ← ScrivereVettoreGenere(vettoreGenere, BaseDati)

vettoreBranoGenere ← LeggereVettoreBranoGenere(BaseDati)
j ← 0
esito ← falso
Mentre(j < ELEM_MAX) AND (esito = falso)
    posizioneCollegamento ← RicercaPosizioneBranoGenereIdGenere(idGenereDaCancellare,
                                                                    vettoreBranoGenere)

    Se(posizioneCollegamento <> SENTINELLA)
        Allora
            BaseDati ← CancellareBrano(BaseDati, campo idBrano di posizioneCollegamento-esimo elemento di
                                                                    vettoreBranoGenere)
            vettoreBranoGenere ← SpostareSinistraVettoreBranoGenere(vettoreBranoGenere, posizioneCollegamento)
            BaseDati ← ScrivereVettoreBranoGenere(vettoreBranoGenere, BaseDati)
        Altrimenti
            stampa("Genere non collegato a nessun brano")
            esito ← vero
    FINE
    j ← j+1
FINE
FINE

```

FUNZIONE “CancellareBrano”

INPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi
idBranoDaCancellare	tipo intero	valore indicante l'id della struttura di tipo Brano da cancellare

OUTPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema aggiornate
----------	---------------	---

DATI DI LAVORO

esito	tipo booleano	“vero” se ho correttamente eliminato il brano e i suoi collegamenti ad altre strutture, “falso” altrimenti
vettoreBrano	tipo vettore Brano	vettore dei brani del database
vettoreBranoArtista	tipo vettore BranoArtista	vettore dei collegamenti BranoArtista del database
vettoreBranoAlbum	tipo vettore BranoAlbum	vettore dei collegamenti BranoAlbum del database
vettoreBranoPlaylist	tipo vettore BranoPlaylist	vettore dei collegamenti BranoPlaylist del database
i	tipo intero >= 0 e <= ELEM_MAX	valore indicante la posizione nel vettoreBrano di BranoDaEliminare
j	tipo intero >= 0 e <= ELEM_MAX	valore indicante la j-esima iterazione alla ricerca di un collegamento brano-album o brano -artista nel vettoreBranoAlbum o vettoreBranoArtista
posizioneCollegamento	tipo intero >= 0 e <= ELEM_MAX	valore indicante la posizione nella tabella di un collegamento brano-artista o brano-album

PSEUDOCODICE

```

vettoreBrano ← LeggereVettoreBrano(BaseDati)

i ← 0
Mentre(LeggereIdBrano(LeggereBrano(vettoreBrano,i)) <> idBranoDaCancellare) AND (i < ELEM_MAX){
    i ← i+1
}
FINE
Se(i = ELEM_MAX)
    Allora stampa("Brano selezionato non esistente")
Altrimenti

```

```

vettoreBrano ← SpostareSinistraVettoreBrano(vettoreBrano, i)
BaseDati ← ScrivereVettoreBrano(vettoreBrano, BaseDati)

vettoreBranoArtista ← LeggereVettoreBranoArtista(BaseDati)
j<-0
esito ← falso
Mentre(j<ELEM_MAX)AND(esito = falso)
    posizioneCollegamento ← RicercaPosizioneBranoArtistaIdBrano(idBranoDaCancellare, vettoreBranoArtista)
    Se(posizioneCollegamento <> SENTINELLA)
        Allora
            vettoreBranoArtista ← SpostareSinistraVettoreBranoArtista(vettoreBranoArtista, posizioneCollegamento)
            BaseDati ← ScrivereVettoreBranoArtista( vettoreBranoArtista, BaseDati)
        Altrimenti
            stampa(“Brano non collegato a nessun artista”)
            esito ← vero
    FINE
    j ← j+1
FINE

j ← 0
esito ← falso
Mentre(j<ELEM_MAX)AND(esito = falso)
    vettoreBranoAlbum ← LeggereVettoreBranoAlbum(BaseDati)
    posizioneCollegamento ← RicercaPosizioneBranoAlbumIdBrano(idBranoDaCancellare, vettoreBranoAlbum)
    Se(posizioneCollegamento <> SENTINELLA)
        Allora
            vettoreBranoAlbum ← SpostareSinistraVettoreBranoAlbum(vettoreBranoAlbum,posizioneCollegamento)
            BaseDati ← ScrivereVettoreBranoAlbum( vettoreBranoAlbum, BaseDati)
        Altrimenti
            stampa(“Brano non collegato a nessun album”)
            esito ← vero
    FINE
    j ← j+1
FINE

j ← 0
esito ← falso
Mentre(j<ELEM_MAX)AND(esito = falso)
    vettoreBranoPlaylist ← LeggereVettoreBranoPlaylist(BaseDati)
    posizioneCollegamento ← RicercaPosizioneBranoPlaylistIdBrano(idBranoDaCancellare, vettoreBranoPlaylist)
    Se(posizioneCollegamento <> SENTINELLA)
        Allora
            vettoreBranoPlaylist ← SpostareSinistraVettoreBranoPlaylist(vettoreBranoPlaylist,posizioneCollegamento)
            BaseDati ← ScrivereVettoreBranoPlaylist( vettoreBranoPlaylist, BaseDati)
        Altrimenti
            stampa(“Brano non collegato a nessuna playlist”)
            esito ← vero
    FINE
    j ← j+1
FINE
FINE

```

FINE

FUNZIONE “CancellareBranoDaPlaylist”

INPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi
idBranoDaCancellare	tipo intero	valore indicante l’id della struttura di tipo Brano da scollegare dalla playlist
idPlaylistDaAggiornare	tipo intero	valore indicante l’id della struttura di tipo Playlist da cui scollegare un brano

OUTPUT

musica tipo Database struttura contenente tutte le informazioni del sistema aggiornate

DATI DI LAVORO

vettoreBranoPlaylist tipo vettore BranoPlaylist vettore dei collegamenti BranoPlaylist del database
posizioneCollegamento tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$ valore indicante la posizione nella tabella di un collegamento brano-playlist

PSEUDOCODICE

```
vettoreBranoPlaylist ← LeggereVettoreBranoPlaylist(BaseDati)
posizioneCollegamento ← RicercaPosizioneBranoPlaylistIdBrano(idBranoDaCancellare, vettoreBranoPlaylist)
Se(posizioneCollegamento <> SENTINELLA)
Allora    vettoreBranoPlaylist ← SpostareSinistraVettoreBranoPlaylist(vettoreBranoPlaylist, posizioneCollegamento)
          BaseDati ← ScrivereVettoreBranoPlaylist(vettoreBranoPlaylist, BaseDati)
Altrimenti stampa("Brano non collegato alla playlist")
FINE
```

FUNZIONE “CancellareAlbum”

INPUT

BaseDati tipo Database struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi

idAlbumDaCancellare tipo intero valore indicante l'id della struttura di tipo Album da cancellare

OUTPUT

musica tipo Database struttura contenente tutte le informazioni del sistema aggiornate

DATI DI LAVORO

esito tipo booleano “vero” se ho correttamente eliminato l’album e i suoi collegamenti ad altre strutture, “falso” altrimenti
vettoreAlbum tipo vettore Album vettore dei album del database
vettoreBranoAlbum tipo vettore BranoAlbum vettore dei collegamenti BranoAlbum del database
i tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$ valore indicante la posizione nel vettoreAlbum di BranoDaEliminare
j tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$ valore indicante la j-esima iterazione alla ricerca di un collegamento brano-album nel vettoreBranoAlbum
posizioneCollegamento tipo intero ≥ 0 e $\leq \text{ELEM_MAX}$ valore indicante la posizione nella tabella di un collegamento brano-album

PSEUDOCODICE

```
vettoreAlbum ← LeggereVettoreAlbum(BaseDati)
```

```
i ← 0
```

```
Mentre(LeggereIdAlbum(LeggereAlbum(vettoreAlbum,i)) <> idAlbumDaCancellare)AND(i < ELEM_MAX){  
    i ← i+1
```

```
FINE
```

```
Se(i = ELEM_MAX)
```

```
Alora stampa("Album selezionato non esistente")
```

```
Altrimenti
```

```
    vettoreAlbum ← SpostareSinistraVettoreAlbum(vettoreAlbum, i)  
    BaseDati ← ScrivereVettoreAlbum(vettoreAlbum, BaseDati)
```

```
    j ← 0
```

```
    esito ← falso
```

```
    Mentre(j < ELEM_MAX)AND (esito = falso)
```

```
        vettoreBranoAlbum ← LeggereVettoreBranoAlbum(BaseDati)
```

```
        posizioneCollegamento ← RicercaPosizioneBranoAlbumIdAlbum(idAlbumDaCancellare, vettoreBranoAlbum)
```

```

    Se(posizioneCollegamento <> SENTINELLA)
    Allora
        vettoreBranoAlbum ← SpostareSinistraVettoreBranoAlbum(vettoreBranoAlbum,posizioneCollegamento)
        BaseDati ← ScrivereVettoreBranoAlbum( vettoreBranoAlbum, BaseDati)
    Altrimenti
        stampa(“Album non collegato a nessun brano”)
        esito ← vero
    FINE
j ← j+1
FINE

```

FINE

FUNZIONE “CancellareArtista”

INPUT

BaseDati	tipo Database	struttura contenente tutte le informazioni del sistema, nel quale cancellare uno o più elementi
idArtistaDaCancellare	tipo intero	valore indicante l’id della struttura di tipo Artista da cancellare

OUTPUT

musica	tipo Database	struttura contenente tutte le informazioni del sistema aggiornate
--------	---------------	---

DATI DI LAVORO

esito	tipo booleano	“vero” se ho correttamente eliminato l’artista e i suoi collegamenti ad altre strutture, “falso” altrimenti
vettoreArtista	tipo vettore Artista	vettore dei brani del database
vettoreBranoArtista	tipo vettore BranoArtista	vettore dei collegamenti BranoArtista del database
i	tipo intero>=0 e <=ELEM_MAX	valore indicante la posizione nel vettoreArtista di ArtistaDaCancellare
j	tipo intero>=0 e <=ELEM_MAX	valore indicante la jesima iterazione alla ricerca di un collegamento brano-artista nel vettoreBranoArtista
posizioneCollegamento	tipo intero>=0 e <=ELEM_MAX	valore indicante la posizione nella tabella di un collegamento brano-artista

PSEUDOCODICE

```

vettoreArtista ← LeggereVettoreArtista(BaseDati)
ArtistaDaCancellare ← RicercareArtistaId(vettoreArtista, idArtistaDaCancellare)

i ← 0
Mentre(LeggereIdArtista(LeggereArtista(vettoreArtista,i)) <> idArtistaDaCancellare)AND(i < ELEM_MAX){
    i ← i+1
}
FINE
Se(i = ELEM_MAX)
Allora stampa(“Artista selezionato non esistente”)
Altrimenti
    vettoreArtista ← SpostareSinistraVettoreArtista(vettoreArtista, i)
    BaseDati ← ScrivereVettoreArtista(vettoreArtista, BaseDati)

    vettoreBranoArtista ← LeggereVettoreBranoArtista(BaseDati)
    j ← 0
    esito ← falso
    Mentre(j<ELEM_MAX)AND(esito = falso)
        posizioneCollegamento ← RicercaPosizioneBranoArtistaIdArtista(idArtistaDaCancellare, vettoreBranoArtista)
        Se(posizioneCollegamento <> SENTINELLA)
            Allora
                vettoreBranoArtista ← SpostareSinistraVettoreBranoArtista(vettoreBranoArtista, posizioneCollegamento)
                BaseDati ← ScrivereVettoreBranoArtista( vettoreBranoArtista, BaseDati)

```

```

        Se(ContareCollegamentiBranArtistaIdBran(campo idBran di posizioneCollegamento-esimo elemento
        di vettoreBranArtista, vettoreBranArtista) = 0)
        Allora BaseDati ← CancellareBran(BaseDati, campo idBran di posizioneCollegamento-esimo
        elemento di vettoreBranArtista)
        //sto cancellando dal sistema i brani senza artista
        FINE
    Altrimenti
        stampa(“Artista non collegato a nessun brano”)
        esito ← vero
    FINE
    j ← j+1
FINE

```

FINE

FUNZIONE “SpostareSinistraVettoreGENERICO”

INPUT

vet	tipo Vettore	vettore di qualsiasi tipo
i	tipo intero	valore indicante la posizione del vettore dalla quale spostare a sinistra gli elementi

OUTPUT

vet	tipo Vettore	vettore di qualsiasi tipo i cui elementi sono stati traslati a sinistra a partire dalla posizione i
-----	--------------	---

PSEUDOCODICE

```

Mentre(i < ELEM_MAX -1)
    iesimo elemento di vet ← (i+1)esimo elemento di vet
    i ← i+1
FINE

```

FUNZIONE “ContareCollegamentiBranArtistaIdBran”

INPUT

idBran	tipo intero	valore indicante il brano collegato a un artista da cercare
vettoreBranArtista	tipo vettore di BranArtista	vettore contenente i collegamenti brano-artista di BaseDati

OUTPUT

cont	tipo intero ≥ 0	valore indicante il numero di occorrenze di idBran
------	-----------------	--

DATI DI LAVORO

i	tipo intero ≥ 0	valore indicante la posizione nel vettoreBranAlbum considerata
---	-----------------	--

PSEUDOCODICE

```

i ← 0
cont ← 0
Mentre(i < ELEM_MAX)
    Se(campo idBran di iesimo elemento di vettoreBranArtista = idBran)
        Allora cont ← cont + 1
    FINE
    i ← i+1
FINE

```
