

Una radio vuole gestire una sua hit parade dei brani musicali più richiesti nell'arco di una settimana dai suoi radioascoltatori. Per ogni brano, viene memorizzato il titolo, l'autore (che si suppone per semplicità unico) e il numero delle richieste pervenute nell'ultima settimana. Di settimana in settimana il numero dei brani presenti nella hit parade può variare in quanto nuovi brani possono essere stati richiesti oppure brani presenti nella hit parade non essere stati più richiesti. La hit parade è ordinata per valori decrescenti del numero delle richieste.

Le operazioni che possono essere effettuate sul tipo Parade sono le seguenti:

- **Parade p()**  
Costruttore che inizializza la hit parade in modo tale che non risultino brani al suo interno;
- **~Parade()**  
Il distruttore che libera tutta la memoria occupata dalla hit parade;
- **p.inserisci(tit, aut, ric)**  
Operazione che inserisce il brano individuato dal titolo **tit**, dall'autore **aut** e dal numero di richieste **ric** nella hit parade in modo tale da mantenerla ordinata per valori decrescenti del campo **ric**. Si supponga che il brano non sia presente nella hit parade.
- **cout << p**  
L'operatore di uscita per il tipo **Parade**. Per ogni brano, viene stampato il titolo, l'autore ed il numero di richieste. Il formato dell'uscita è il seguente:  
Titolo - Autore - Richieste  
La casa del sole - Pooh - 300  
A te - Jovannotti - 234  
Suicidio d'amore - Nannini - 200
- **Parade p2(p1)**  
Costruttore di copia.
- **p[i]**  
L'operatore di selezione che stampa su uscita standard il titolo, l'autore e il numero di richieste del brano in posizione **i** nella hit parade (si consideri che il primo brano è in posizione 0). Se non esiste nessun brano in posizione **i** l'operatore non stampa nulla. L'operatore restituisce **true** se la posizione individua un brano, altrimenti restituisce **false**.
- **p.rimuovi(ric)**  
Operazione che rimuove dalla hit parade tutti i brani con un numero di richieste inferiori a **ric**. La funzione restituisce il numero di richieste eliminate.

Utilizzando il linguaggio C++, realizzare il tipo di dati astratti Parade, definito dalle precedenti specifiche.

## NOTE SULLO SVOLGIMENTO DELLA PROVA PRATICA:

- Effettuare il **login**  
Nome: studenti  
Password: studenti
- Aprire il *Dev-C++* (dal Menù *Avvio* (o *Start*) nella barra degli strumenti in fondo allo schermo, selezionare *Programmi* e quindi *Dev-C++*)
- **Prima di iniziare a svolgere l'elaborato**, selezionare la voce *Identifica studente* nel menù *Strumenti* all'interno dell'ambiente *Dev-C++* e inserire i dati richiesti
- Dal menu *File* del *Dev-C++*, aprire il progetto *esainf.dev* presente nella cartella *c:\esame\esaInf*. Il progetto contiene tre file, denominati *compito.h*, *compito.cpp* e *main.cpp*
- Scrivere la dichiarazione della classe nel file *compito.h* e la definizione delle funzioni nel file *compito.cpp*. Il file *main.cpp* contiene la funzione principale *main()* che serve a verificare le funzioni scritte nel file *compito.cpp*. Il file *main.cpp* può essere modificato. **Si tenga presente, comunque, che in sede di valutazione dell'elaborato verrà considerato esclusivamente il contenuto dei file *compito.h* e *compito.cpp***
- **Per una corretta stampa dell'elaborato**, non scrivere righe di codice di lunghezza eccessiva (mantenersi entro i margini imposti dall'ambiente *Dev-C++* cioè entro la linea verticale presente alla destra della pagina);
- **Per la Consegna:**
  - Selezionare la voce *Consegna* dal menù *Strumenti* (o *Tools*) all'interno dell'ambiente *Dev-C++* e premere il tasto INVIO fino a quando non viene chiusa la finestra che è stata attivata.
  - Aspettare al proprio posto di essere chiamati per verificare la stampa del proprio elaborato, firmarlo e **consegnarlo definitivamente**.

**Condizione necessaria** per la correzione dell'elaborato è che le prime tre funzioni siano state implementate correttamente.