

Un'università vuole realizzare un sistema per la gestione dell'assegnazione delle borse di studio. Per ogni studente, il sistema memorizza la matricola (rappresentata da una stringa), il numero di anni trascorsi dall'anno di immatricolazione, il numero di esami ed il reddito. Nel sistema, gli studenti sono memorizzati in modo tale da risultare ordinati per redditi crescenti e, in caso di redditi uguali, per valori decrescenti del rapporto tra il numero di esami ed il numero di anni trascorsi dall'anno di immatricolazione.

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

- **`GestBorse g()`**
Costruttore che inizializza il gestore in modo tale che non risultino studenti;
- **`GestBorse g(nomeFile)`**
Costruttore che inizializza il gestore con il contenuto del file **`nomeFile`**; il file è composto da righe contenenti una stringa, due numeri interi ed un numero reale, che rappresentano, rispettivamente, la matricola, il numero di anni trascorsi dall'anno di immatricolazione, il numero di esami ed il reddito. Il file viene letto fino alla fine e gli studenti vengono inseriti nel sistema di gestione in modo tale da risultare ordinati come descritto sopra;
- **`~GestBorse ()`**
Il distruttore che libera tutta la memoria occupata dal gestore;
- **`g.inserisci(mat, numAnni, numEs, red)`**
Operazione che inserisce lo studente nel sistema di gestione in modo tale che gli studenti risultino ordinati. Si supponga che lo studente non sia presente nel sistema.
- **`cout << g`**
L'operatore di uscita per il tipo **`GestBorse`**. Per ogni studente, viene stampato la matricola, il numero di esami, gli anni trascorsi dall'anno di immatricolazione ed il reddito. Il formato dell'uscita è il seguente:

Matr	Esami	Anni	Reddito
d78963	8	2	8000
b5678	3	1	9000
a1234	12	4	10000
c5896	2	1	10000
- **`GestBorse g2(g1)`**
Costruttore di copia.
- **`g1+g2`**
L'operatore di somma che restituisce un sistema di gestione ordinato contenente l'unione degli studenti presenti in **`g1`** e **`g2`** (si supponga che non ci siano studenti con la stessa matricola).
- **`g.verifica(mat, red)`**
Operazione che cambia il reddito dello studente individuato dalla matricola **`mat`** e riposiziona lo studente in modo tale che gli studenti risultino ordinati. La funzione restituisce **`false`** se lo studente non è presente; altrimenti restituisce **`true`**.

Utilizzando il linguaggio C++, realizzare il tipo di dati astratti `GestBorse`, 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.