

Esame di Programmazione II

Appello di giorno 23 Giugno 2017
Università degli Studi di Catania - Corso di Laurea in Informatica

Testo della Prova

Definizione Iniziale.

Ad un concorso pubblico un candidato viene identificato dal suo *codice fiscale*. Dopo aver sostenuto l'esame, per stilare la graduatoria finale, la commissione tiene in considerazione la valutazione complessiva, il numero di domande a cui il candidato ha risposto correttamente e, infine, l'età del candidato.

Dato il candidato C_i , indichiamo con v_i la sua valutazione complessiva, con r_i il numero di domande a cui il candidato a risposto correttamente e con e_i l'età del candidato.

In base alle direttive del bando di concorso un candidato C_i supera in graduatoria un candidato C_j (quindi $C_i < C_j$) se:

- $v_i > v_j$;
- $v_i = v_j$ e $r_i > r_j$;
- $v_i = v_j$, $r_i = r_j$ e $e_i < e_j$.

1. Classe Candidato (10 punti)

Si implementi in C++ la classe **Candidato** fornendo, oltre al costruttore, dei metodi per la modifica e la valutazione dei valori v (valutazione complessiva), r (numero di risposte corrette) ed e (età). Si effettui inoltre l'overloading degli operatori di confronto ($<$, $>$, $=$, $!$, $<=$, $>=$) in modo da poter confrontare istanze di questa classe.

2. Classe Lista (12 punti)

Si implementi in C++ la classe **Lista** rappresentante un insieme di elementi implementato mediante una lista singolarmente linkata ed ordinata (con ordine non decrescente). Implementare la lista utilizzando i Template. La classe deve permettere le seguenti operazioni:

- la ricerca di un elemento dell'insieme sulla base della sua chiave;
- l'inserimento di un nuovo elemento;
- la cancellazione di un elemento;
- la stampa degli elementi presenti nell'insieme.

3. Classe Graduatoria (8 punti)

Si implementi in C++ la classe **Graduatoria** rappresentante l'insieme dei candidati che hanno preso parte al concorso. L'insieme deve essere implementato mediante una lista singolarmente linkata ed ordinata (con ordine non decrescente), in base ai criteri forniti in precedenza. La classe deve permettere le seguenti operazioni:

- l'inserimento di un nuovo candidato alla graduatoria;
- la ricerca di un candidato sulla base della sua posizione;
- la stampa della graduatoria di tutti i candidati che hanno preso parte al concorso.

Test e valutazione del proprio elaborato.

Per la verifica e la valutazione del proprio elaborato si eseguano le seguenti istruzioni:

```
// esercizio n.1
int v=89, r=60, e=27;
Candidato *a = new Candidato("GTRSDF94S03C351J", v, r, e);
Candidato *b = new Candidato("FGTSER95Y05H351T", 63, 63, 24);
Candidato *c = new Candidato("RYUASD93R12E353H", 89, 60, 25);
if( *a < *b) cout << "True"; else cout << "False";
if( *a >= *b) cout << "True"; else cout << "False";
if( *a == *c) cout << "True"; else cout << "False";
if( *a <= *c) cout << "True"; else cout << "False";

// esercizio n.2
Lista<int> *l = new Lista<int>();
l->ins(2)->ins(3)->ins(12)->ins(4)->ins(9)->ins(5);
l->del(3)->del(6);
l->print();

// esercizio n.3
Graduatoria *g = new Graduatoria();
g->ins(*a)->ins(*b)->ins(*c);
cout << g->search(2);
g->print();
```

L'output del programma sarà quindi:

```
True
False
False
False
12, 9, 5, 4, 2
RYUASD93R12E353H, GTRSDF94S03C351J, FGTSER95Y05H351T
```