**Esercizi**

**1**.

Riferimenti: Listato 15.1,

**Soluzione**: Si veda il codice dei file sorgente ListaConcatenataDiStringhe e ListaConcatenataDiStringheDemo.java

**2**.

**Soluzione**: Si veda il codice del file sorgente NodoDuale.java

**3**.

**Soluzione**: Si veda l’immagine Esercizio1.jpg

**4**.

**Riferimenti:** Listato 15.16

**Soluzione**: Si veda il frammento di codice del file sorgente ListaConcatenataDiStringheConIteratoreFrammento.java

**5**.

**Riferimenti:** Listato 15.16

**Soluzione**: Si veda il frammento di codice del file sorgente ListaConcatenataDiStringheConIteratoreFrammento.java

**6**.

**Riferimenti:** Listato 15.16

**Soluzione**: Si veda il frammento di codice del file sorgente ListaConcatenataDiStringheConIteratoreFrammento.java

**8**.

**Riferimenti:** Listato 15.8

**Soluzione**: Si veda il frammento di codice del file sorgente Incassi.java

\*\*\*\*\*\*

**Programmi**

**1.**

Definisce una *inner class* chiamata NodoLista in cui il dato è di tipo Specie.

**Riferimenti:** Listato 15.4, Listato 8.16

**Soluzione**: Si veda il codice dei file sorgente ListaConcatenataDiSpecieAutoContenuta.java e ListaConcatenataDiSpecieAutoContenutaDemo.java

**2**.

Il progetto è simile al progetto 1 di questo capitolo. Definisce una *inner class* chiamata NodoLista in cui il dato è di tipo Dipendente.

**Riferimenti:** Progetto 7 del capitolo 13, Listato 15.4

**Soluzione**: Si veda il codice dei file sorgente ListaConcatenataDiDipendentiAutoContenuta.java e ListaConcatenataDiDipendentiAutoContenutaDemo.java

**3.**

**Riferimenti:** Listato 15.19

**Soluzione**: Si veda il codice del file sorgente Insieme.java e InsiemeDemo.java

\*\*\*\*\*\*

**Progetti**

**1.**

Questo programma è basato sul codice del listato 15.6. La soluzione qui fornita include il supporto per l'iterazione, sia in avanti che indietro. Sia l'iterazione avanti sia quella indietro utilizza il metodo ancoraDaIterare, ma l'iterazione indietro utilizza il reimpostaIteratoreAlContrario piuttosto che reimpostaIterazione. La classe NodoLista definisce una variabile di istanza precedente che rende superflua la definizione di una ulteriore variabile precedente nella classe esterna. Per rendere l'iterazione indietro più facile da scrivere, c'è un riferimento alla coda della lista così come all'inizio della lista. Un metodo aggiuntivo, cercaNellaLista, cerca un elemento nella lista e imposta corrente a quell'elemento se viene trovato. Se non viene trovato, corrente è impostato su null. Il metodo mostraStatoLista ha lo scopo di verifica e stampa la testa della lista, l'elemento corrente, la coda e il numero di elementi nella lista.

**Riferimenti**: Listato 15.6

**Soluzione**: Si veda il codice dei file sorgente ListaConcatenataDoppia.java e ListaConcatenataDoppiaDemo.java

**2.**

Questo programma implementa una versione semplificata di una lista concatenata che può essere utilizzata per rilevare cicli mentre si costruisce un minimo albero ricoprente. Per essere veramente efficiente, si dovrebbe usare una tabella hash invece di un ArrayList per memorizzare gli elementi. Si dovrebbe anche tenere traccia dell'altezza degli alberi e collegare sempre l'albero più corto all'albero più alto.

Questa implementazione utilizza un metodo privato per determinare se l'elemento è quello del gruppo. Il metodo getRappresentativa scorre i puntatori fino a quando non trova null. Il metodo unione effettuerà un collegamento quando entrambi gli argomenti sono nel gruppo e non sono nello stesso insieme. Si noti che una volta che un collegamento è stato creato, non occorre mai cambiarlo.

**Soluzione**: Si veda il codice dei file sorgente ContenitoreGruppi.java

**3.**

La soluzione si basa sulla versione generica della lista collegata riportata nel listato 15.8. La definizione di nodo deve essere cambiata per includere un dato in più (il conteggio). Il metodo di aggiunta deve essere modificato per fare in modo che prima venga effettuata una ricerca e quindi poi venga incrementato il conteggio se l'uccello è già nella lista. Si noti che una volta che un collegamento è impostato, non ha mai bisogno di essere cambiato.

**Riferimenti**: Listato 15.8

**Soluzione**: Si veda il codice del file sorgente StudioUccelli.java

**4.**

**Soluzione**: Si veda il codice dei file sorgente NumeroEnorme.java e NumeroEnormeDemo.java

**6.**

Questo programma è simile alla lista generica implementata nel listato 15.8. Il codice è stato leggermente modificato per chiamare il metodo aggiungiInOrdine e per inserire i dati non in ordine per illustrare che vengono aggiunti invece in maniera ordinata

**Riferimenti**: Listato 15.8

**Soluzione**: Si veda il codice dei file sorgente Elemento.java, ListaConcatenataProgetto6.java, e ListaConcatenataProgetto6Demo.java

**7.**

Questo progetto può essere risolto modificando TabellaHash del Listato 15.17, usando la classe ListaConcatenata del Listato 15.8 e effettuando l’override del metodo equals di Object nella classe Dipendente del Progetto 7 del Capitolo 13. Due dipendneti sono uguali se hanno lo stesso nome. Nella classe ListaConcatenata, aggiunto il metodo getDati che restituisce i dati di un nodo specificato. Andrebbe aggiunto il metodo toString alla classe Dipendente.

**Riferimenti**: Listato 15.17, Progetto 7 del Capitolo 13, Listato 15.8.

**Soluzione**: Si veda il codice dei file sorgente TabellaHash.java e TabellaHashDemo.java

**8.**

La soluzione prevede la revisione della classe Insieme in modo da usare una hash table come struttura dati interna.

Si è provveduto a rimuovere la definizione della classe Nodo nella nuova classe Insieme e viene usata al suo posto la classe NodoLista definita come *inner class* nella classe ListaCollegata del Listato 15.8.

All’interno della nuova classe insieme, è stata rimossa la classe che fungeva da iteratore poiché non necessaria per la risoluzione di questo programma. Sempre all’interno della nuova classe Insieme sono stati rimossi i metodi rimuovi e numeroElementi.

Sono stati aggiunti i metodi di *get* per il collegamento e dati della classe NodoLista. Infine è stato aggiunto il metodo getTesta nella classe ListaConcatenata.

**Riferimenti**: Listato 15.8, Listato 15.19 e Listato 15.20.

**Soluzione**: Si veda il codice del file sorgente InsiemeHash.java

**9.**

**Soluzione**: Si veda il codice dei file sorgente Nodo.java e NodoDemo.java