

Esame di Fondamenti di Informatica L-B Ingegneria Gestionale e dei Processi Gestionali (L-Z)

Appello del 15/12/2008

Esercizio 1 (4 punti)

Discutere il concetto di ereditarietà.

Esercizio 2 (6 punti)

Siano date le seguenti funzioni C:

```
int f(int V[], int M) {
    int i=1, j=0, sum=0;

    for(; j<M; i++) {
        sum+=V[i];
        j=i;
    }
    return sum;
}

void g(int U[], int V[], int N) {
    int j;

    for(j=0; j<N; j++) U[j]=f(V, j+1);
}
```

1. Calcolare la complessità in passi base della funzione `f` nei termini del parametro `M`.
2. Calcolare la complessità in passi base della funzione `g` nei termini del parametro `N`.
3. Calcolare la complessità asintotica della funzione `g` nei termini del parametro `N`.

Esercizio 3 (6 punti)

Nel centro di smistamento regali al polo nord, Babbo Natale si sta preparando per la distribuzione dei doni ai bambini che hanno fatto i bravi durante l'anno. Le informazioni sui regali vengono memorizzate all'interno di un sistema informatico. In particolare, per ogni regalo vengono indicati il nome (es. "Winx", "Gormiti"), il numero di richieste pervenute per tale regalo ed il prezzo unitario.

Si scriva una classe `Regalo` per Babbo Natale che:

1. Possieda un opportuno costruttore con parametri.
2. Presenti opportuni metodi che permettano di accedere alle variabili di istanza della classe.
3. Possieda il metodo `costoTotale` che calcoli il costo totale per le richieste di un certo regalo.
4. Presenti il metodo `toString` che fornisca la descrizione del regalo.
5. Possieda il metodo `equals` per stabilire l'uguaglianza con un altro oggetto `Regalo` (l'uguaglianza va verificata unicamente sul nome).

Esercizio 4 (9 punti)

Si scriva una classe `Magazzino` che memorizzi le informazioni relative ai vari magazzini di stoccaggio regali sparsi per il mondo (e opportunamente nascosti dalla vista dei bambini). In particolare, oltre all'indirizzo del magazzino, i regali ivi contenuti vanno memorizzati all'interno di una lista. La classe `Magazzino` deve inoltre:

1. Possedere un opportuno costruttore (inizialmente il magazzino è vuoto).
2. Possedere un metodo `getIndirizzo` che restituisca l'indirizzo del magazzino.
3. Presentare un metodo `cercaRegalo` che, dato il nome di un regalo, restituisca l'oggetto `Regalo` corrispondente, se questo esiste all'interno del magazzino.
4. Possedere un metodo `aggiungiRegalo` che, dato un oggetto `Regalo`, lo inserisca in testa alla lista, controllando che questa non lo contenga già (suggerimento: si utilizzi il metodo `add(int i, Regalo r)` della classe `List<Regalo>`).
5. Presentare il metodo `piuCostoso` che restituisca il regalo avente il costo totale più elevato, tra quelli presenti nel magazzino.
6. Possedere il metodo `toString` che restituisca una stringa che fornisca una descrizione del magazzino, comprendendo anche tutti i regali ivi contenuti.

Esercizio 5 (6 punti)

Si scriva un'applicazione per Babbo Natale che:

1. Crei un insieme di oggetti `Magazzino`.
2. Crei un oggetto `Magazzino`, lette da tastiera le informazioni necessarie.
3. Provveda ad inserire all'interno dell'insieme di cui al punto 1. l'oggetto creato al punto 2., controllando che un oggetto uguale non esista già.
4. Letto da tastiera il nome di un regalo, stampi a video l'indirizzo di tutti i magazzini che contengono tale regalo.

Per la lettura di dati da tastiera è possibile utilizzare l'oggetto `Lettore.in`, definito all'interno del package `fi.ji.io`, che possiede i seguenti metodi:

- `boolean leggiBoolean()` Legge un boolean (delimitato da spazi).
- `char leggiChar()` Legge un singolo carattere.
- `double leggiDouble()` Legge un numero razionale (delimitato da spazi).
- `float leggiFloat()` Legge un numero razionale (delimitato da spazi).
- `int leggiInt()` Legge un intero (delimitato da spazi).
- `String leggiLinea()` Legge una linea di testo.
- `String leggiString()` Legge una parola senza spazi al suo interno.

Soluzione Esercizio 2

Domanda 1:

2 assegnamenti	3
while	M+1
sum+=V[i]	M
j=i	M
i++	M
Totale	4M+4

Domanda 2:

1 assegnamento	1
for	N+1
chiamata di f	N
complessità di f	$2N^2 + 6N$
j++	N
Totale	$2N^2 + 9N + 2$

Domanda 3:

Complessità asintotica: $O(N^2)$

Soluzione Esercizio 3

```
class Regalo{
    private String nome;
    private int richieste;
    private float prezzo;

    public Regalo(String nome, int richieste, float prezzo) {
        this.nome = nome;
        this.richieste = richieste;
        this.prezzo = prezzo;
    }

    public String getNome() { return nome; }
    public int getRichieste() { return richieste; }
    public float getPrezzo() { return prezzo; }

    public float costoTotale() { return prezzo*richieste; }

    public String toString() {
        return nome + "(" + richieste + "): " + prezzo + "€";
    }

    public boolean equals(Object o) { return equals((Regalo) o); }
    public boolean equals(Regalo r) { return (nome.equals(r.nome)); }
}
```

Soluzione Esercizio 4

```
import java.util.*;
class Magazzino {
    private String indirizzo;
    private List<Regalo> regali;

    public Magazzino(String indirizzo) {
        this.indirizzo = indirizzo;
        regali = new LinkedList<Regalo>();
    }

    public String getIndirizzo() { return indirizzo; }
```

```
public Regalo cercaRegalo(String nome) {
    for(Regalo r:regali)
        if(r.getNome().equals(nome)) return r;
    return null;
}

public boolean aggiungiRegalo(Regalo r) {
    if(regali.contains(r)) return false;
    regali.add(0, r);
    return true;
}

public Regalo piuCostoso() {
    Regalo max=null;
    for(Regalo r:regali)
        if((max==null)|| (r.costoTotale()>max.costoTotale()))
            max=r;
    return max;
}

public String toString() {
    return "Indirizzo: "+indirizzo+"\n"+regali;
}
}
```

Soluzione Esercizio 5

```
import fiji.io.*;
import java.util.*;

class Applicazione {
    public static void main(String[] args) {
        Set<Magazzino> s=new HashSet<Magazzino>(); // domanda 1
        Magazzino m=new Magazzino(Lettore.in.leggiString()); // domanda 2
        s.add(m); // domanda 3
        String nome=Lettore.in.leggiLinea();
        for(Magazzino mag:s) {
            if(mag.cercaRegalo(nome)!=null)
                System.out.println(mag);
        } // domanda 4
    }
}
```