

Università di Roma “La Sapienza”

A.A. 2006-2007

Facoltà di Ingegneria

Corso di Laurea in Ingegneria Informatica

Corso di “PROGETTAZIONE DEL SOFTWARE I”  
(Canale A-L & M-Z)

ESERCITAZIONE 1:  
INTRODUZIONE ALLA PROGETTAZIONE  
Esercizi sulla modularizzazione (soluzione)

<i>Versione definitiva</i>
----------------------------

# 1 Telefono cellulare

## RISPOSTA

La risposta giusta è C. Per comprendere ciò, dobbiamo considerare le sezioni del manuale come moduli e gli argomenti trattati come servizi offerti. La persona che vuole imparare le regole è assimilabile ad un modulo cliente. Le soluzioni A e B prefigurano due moduli server, dei quali almeno uno (quello in cui poniamo le regole sull'immissione del testo) con bassa coesione. Di conseguenza ci sarebbe un alto accoppiamento fra le sezioni sugli SMS e sulla rubrica: nel caso A, ad esempio, una persona che volesse imparare ad usare la rubrica telefonica e fosse interessata ad una particolare combinazione di maiuscole/minuscole (ad es., "Alessandro Rossi ufficio") dovrebbe necessariamente leggere la sezione sugli SMS. La soluzione C garantisce anche maggiore estendibilità: se in future versioni del manuale utente saranno previste altre regole, saranno illustrate nella sezione apposita. Sull'information hiding non possiamo fare alcuna osservazione, mentre un alto interfacciamento esplicito potrebbe essere garantito dall'indice degli argomenti.

# 2 Ufficio postale

## RISPOSTA

### Modulo A: Servizi postali

#### Servizi offerti agli utenti:

- S-A1: ritiro corrispondenza/telegrammi/pacchi in giacenza
- S-A2: spedizione pacchi, telegrammi e corrispondenza raccomandata/assicurata
- S-A3: acquisto francobolli

#### Come gli utenti possono usufruirne:

- S-A1: presentazione avviso e documento del destinatario, oppure di delega, documento del destinatario e documento proprio, contante (per pagamento diritti di giacenza)
- S-A2: presentazione busta/collo da spedire, moduli per spedizione compilati, contante
- S-A3: presentazione contante

#### Struttura interna:

- $n (\geq 1)$  sportelli con operatore abilitato a tutti i servizi e dotato di terminale, deposito per la modulistica necessaria, deposito contante e deposito moduli compilati
- clienti in fila unica (disciplina FIFO), che accedono al primo sportello libero
- depositi corrispondenza in giacenza, condivisi fra gli operatori

### Modulo B: Servizi finanziari

#### Servizi offerti agli utenti:

- S-B1: pagamento bollettino/invio vaglia
- S-B2: ritorsione vaglia
- S-B3: apertura conto corrente o libretto postale
- S-B4: versamento su conto corrente o libretto postale

#### Come gli utenti possono usufruirne:

- S-B1: presentazione contante/bancomat e moduli compilati
- S-B2: presentazione vaglia e documento del beneficiario, oppure di delega, documento del beneficiario e documento proprio

- S-B3: presentazione documento e codice fiscale dell'intestatario (di entrambi gli intestatari se cointestato) e dei moduli compilati
- S-B4: presentazione contante/assegni e documento dell'intestatario, oppure di delega, documento dell'intestatario e documento proprio, e dei moduli compilati

**Struttura interna:**

- $n (\geq 1)$  sportelli con operatore abilitato a tutti i servizi e dotato di terminale, deposito per la modulistica necessaria, deposito contante e assegni e deposito moduli compilati
- clienti in fila unica (disciplina FIFO), che accedono al primo sportello libero

**Modulo C: Gestione investimenti**

**Servizi offerti agli utenti:**

- S-C1: acquisto titoli (ad es., buoni postali) e fondi di investimento

**Come gli utenti possono usufruirne:**

- S-C1: colloquio con funzionario, seguito da compilazione moduli, seguito da verifica disponibilità finanziaria

**Struttura interna:**

- 1 sportello con funzionario abilitato a tutti i servizi e dotato di deposito per la modulistica necessaria e deposito moduli compilati
- clienti in fila unica (disciplina FIFO), che accedono allo sportello quando libero

**Modulo D: Vendita cancelleria al pubblico**

**Servizi offerti agli utenti:**

- S-D1: acquisto cancelleria

**Come gli utenti possono usufruirne:**

- S-D1: presentazione contante e richiesta beni desiderati

**Struttura interna:**

- 1 sportello con operatore abilitato a tutti i servizi e dotato di deposito cancelleria e deposito contante
- clienti in fila unica (disciplina FIFO), che accedono allo sportello quando libero

**Modulo E: Logistica**

**Servizi offerti agli utenti:**

- S-E1: Disposizione modulistica negli appositi espositori

**Come gli utenti possono usufruirne:**

- S-E1: Ritiro moduli da espositori

**Servizi agli altri moduli:**

- S-E2: Deposito corrispondenza in arrivo (al modulo A)
- S-E3: Ritiro corrispondenza in giacenza e non ritirata entro i termini di legge (al modulo A)
- S-E4: Distribuzione modulistica (ai moduli A-D)
- S-E5: Ritiro contante (ai moduli A, B, D) e assegni (al modulo B)
- S-E6: Ritiro moduli compilati (ai moduli A-C)

## Modulo F: Direzione

### Servizi offerti agli utenti:

- Nessun servizio al pubblico

### Servizi agli altri moduli:

- Chiarimenti su attuazione norme

NOTA: Per erogare il servizio S-C1, il funzionario deve verificare la disponibilità finanziaria, ma non è dotato di terminale. Si dovrà rivolgere ad un operatore del modulo B (accoppiamento). Si può migliorare la modularizzazione dando al funzionario un terminale.

## 3 Funzione JAVA per ricerca elemento in lista

### RISPOSTA

Nel caso in cui si eliminasse l'uso della variabile globale `beta`, l'interfacciamento esplicito della funzione aumenterebbe, e il grado di accoppiamento con gli altri moduli diminuirebbe.

## 4 Funzione JAVA per il calcolo del fattoriale

### RISPOSTA

La funzione entra in un ciclo infinito se viene invocata con un valore negativo per il parametro attuale. Il grado di accoppiamento con altri moduli è alto, proprio perché i moduli clienti, per evitare il rischio del ciclo infinito, sono costretti ad eseguire un test prima di invocare la funzione `Fatt()`, come mostra l'esempio seguente.

```
// File Fattoriale/Cattiva/Client.java
```

```
public class Client {
    public static void main(String[] args) {
        System.out.print("Inserisci un intero: ");
        int my_int = InOut.readInt();
        if (my_int >= 0) // ALTO ACCOPPIAMENTO: il cliente deve
                        // controllare che l'intero non sia negativo
            System.out.print("Il fattoriale di " + my_int + " vale " +
                             Server.Fatt(my_int) + "\n");
    }
}
```

Il grado di coesione della funzione aumenterebbe se la verifica del valore non negativo del parametro fosse eseguita dalla funzione stessa. D'altra parte, inserire la verifica direttamente nella funzione `Fatt()` compromette l'efficienza della funzione, perché ogni chiamata ricorsiva effettuerebbe un test inutile. Ne segue che la qualità del modulo aumenta se la funzione ricorsiva viene nascosta all'esterno rendendola privata, e la funzione che effettua il servizio per i moduli clienti esegue il test e chiama la funzione ricorsiva solo in condizioni corrette:

```
// File Fattoriale/Buona/Server.java
```

```
public class Server {
    private static int FattRic(int n) {
        // precondition: n >= 0
```

```

        if (n == 0)
            return 1;
        else return n * FattRic(n-1);
    }

    public static int Fatt(int n) {
        // restituisce -1 se n < 0
        if (n < 0)
            return -1;
        else return FattRic(n);
    }
}

```

## 5 Ufficio pubblico

### RISPOSTA

L'ufficio pubblico è un modulo ed eroga vari servizi. Il tabaccaio è un altro modulo ed eroga il servizio di vendita delle marche da bollo. Dal punto di vista del cittadino, questi moduli sono molto accoppiati, e quindi la modularizzazione è di bassa qualità (in particolare, il cittadino dovrà spostarsi fisicamente per usufruire dei vari servizi). Una migliore modularizzazione si ottiene permettendo all'ufficio pubblico di vendere le marche da bollo, aumentando così la sua coesione.

## 6 Venditori di computer

### RISPOSTA

Ogni ditta può essere modellata tramite due moduli, ognuno dei quali eroga un solo servizio: la vendita di PC laptop e la vendita di sistemi operativi preinstallati. Nel caso della modularizzazione prefigurata dalla ditta HAL sussiste un forte accoppiamento fra i due moduli, perché il primo servizio implica il secondo e viceversa. Nel caso della ditta Tony l'accoppiamento è minore, perché il secondo servizio implica il primo, ma non viceversa.

## 7 Elenchi telefonici

### RISPOSTA

Ovviamente, per usare l'elenco su CD-ROM dobbiamo avere il computer, che deve essere acceso e disponibile nel momento opportuno. Se assumiamo che il PC con i servizi che esso offre sia un modulo, e che l'elenco telefonico su CD-ROM sia un altro, sussiste accoppiamento fra i due, perché per usufruire del secondo abbiamo bisogno del primo.

In un contesto in cui il computer è sempre acceso e disponibile, ad es., in un ufficio, la ricerca dei dati su CD-ROM sarà tipicamente più rapida (efficiente) della ricerca su elenco cartaceo.

## 8 Dispositivo segnalatore per autobus

### RISPOSTA

I tre display sono servizi offerti. Se li separiamo ci sarà ovviamente alto accoppiamento fra essi (le informazioni visualizzate devono essere le stesse!), con conseguente rischio di incoerenza.

## **9 Saggezza popolare**

### **RISPOSTA**

Pensando che mestolo, secchio, etc. siano servizi, e che il luogo dove si trovano sia un modulo, il proverbio ci esorta a progettare i moduli in maniera tale che abbiano elevata coesione.

## **10 La mia storia**