

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

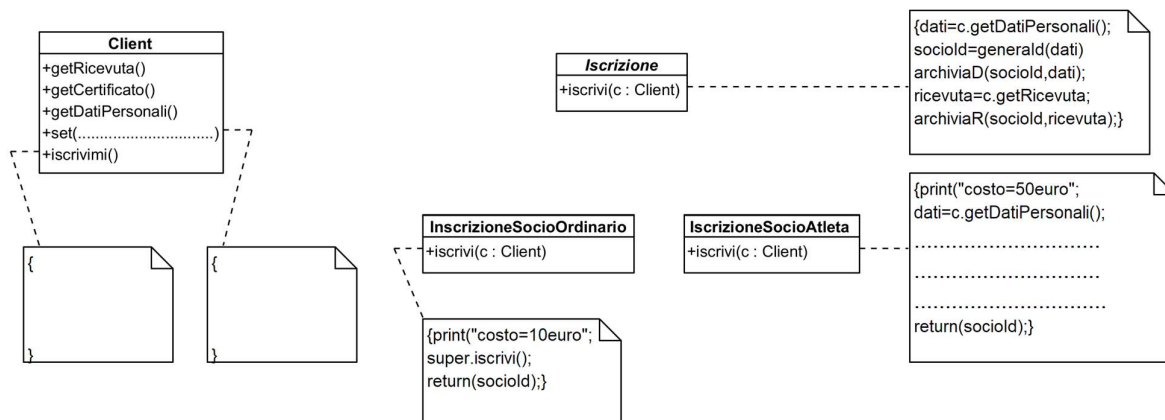
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- a) Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- b) Perché uno stub che risponde in modo random non va bene?

.....

- c) Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

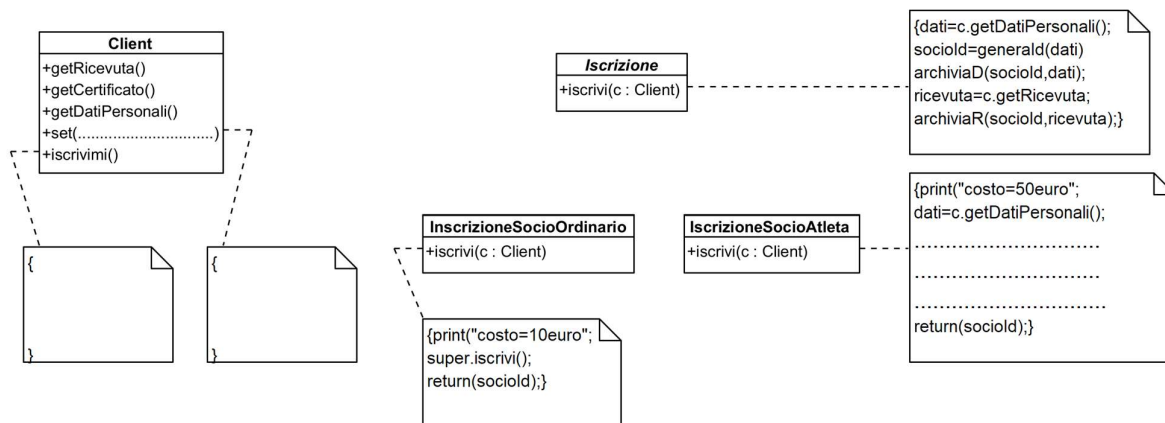
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- a) Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- b) Perché uno stub che risponde in modo random non va bene?

.....

- c) Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

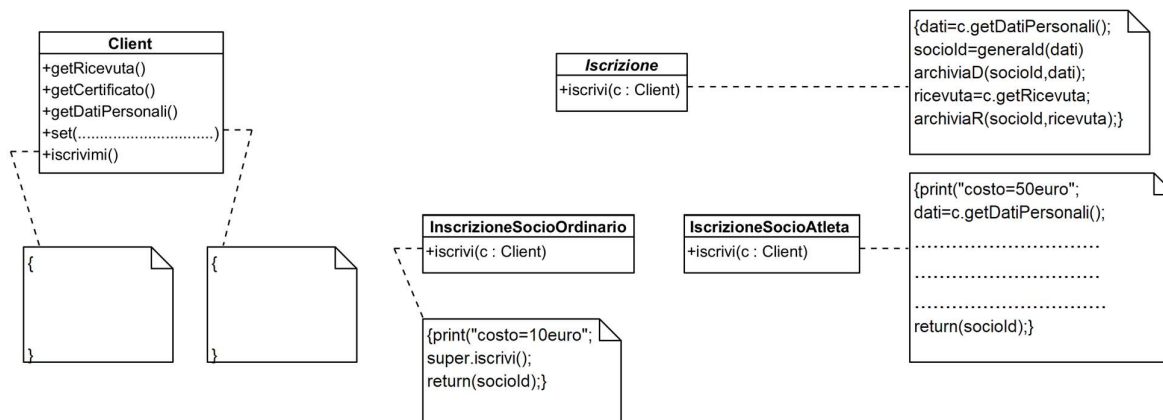
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- a) Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- b) Perché uno stub che risponde in modo random non va bene?

.....

- c) Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

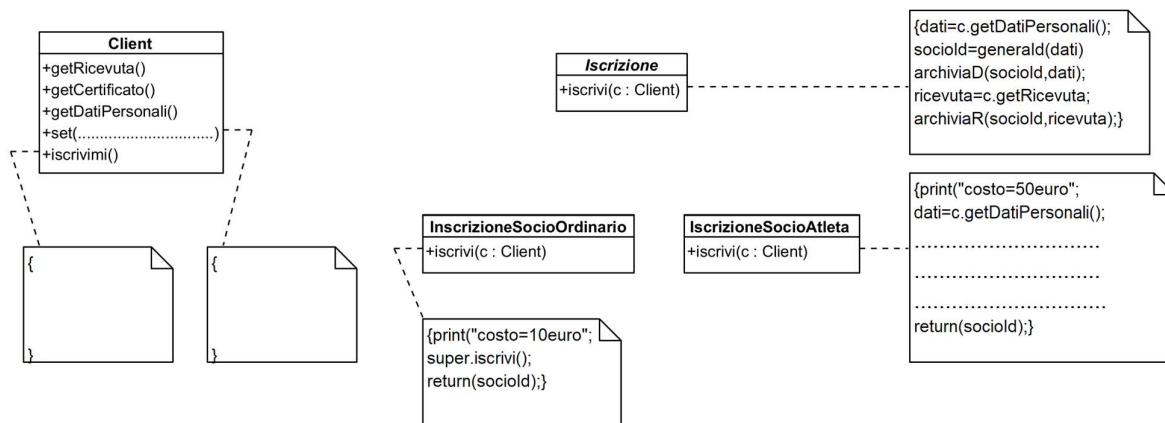
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con “...”

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

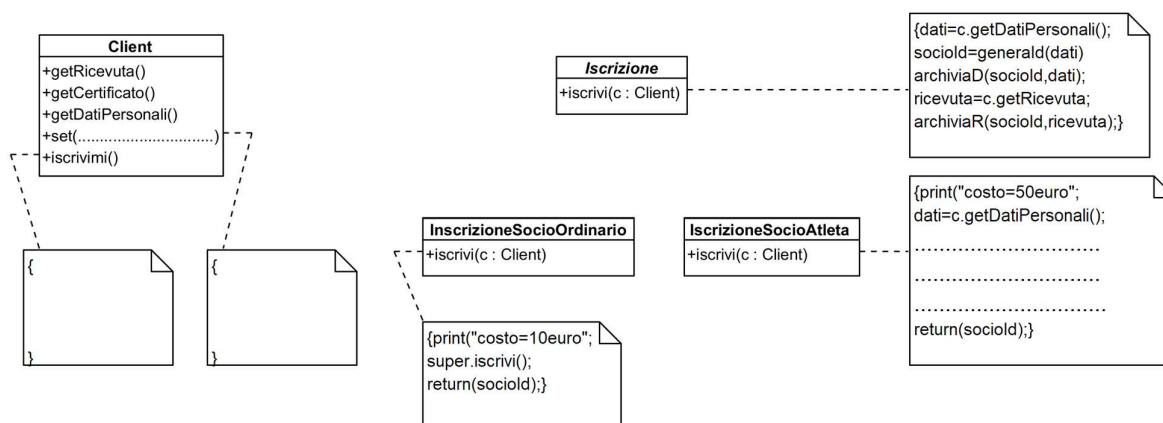
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

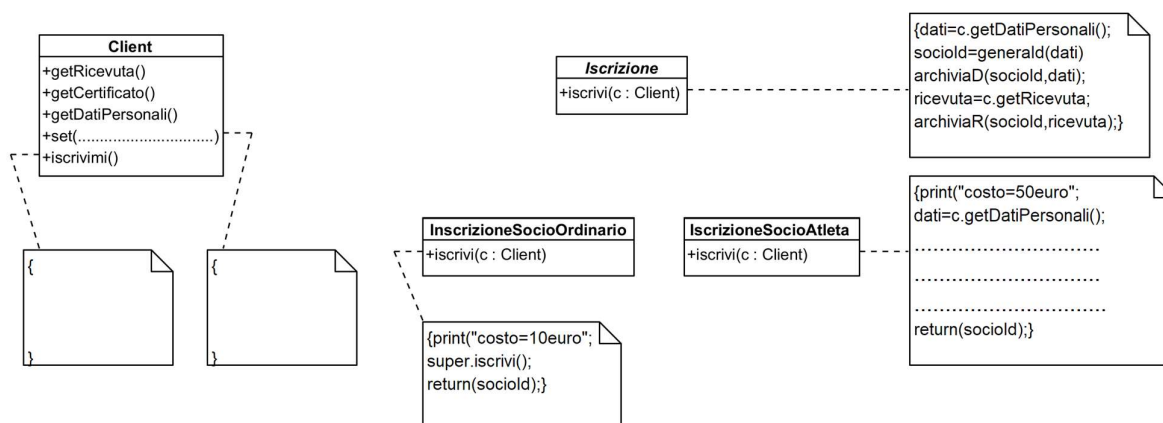
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

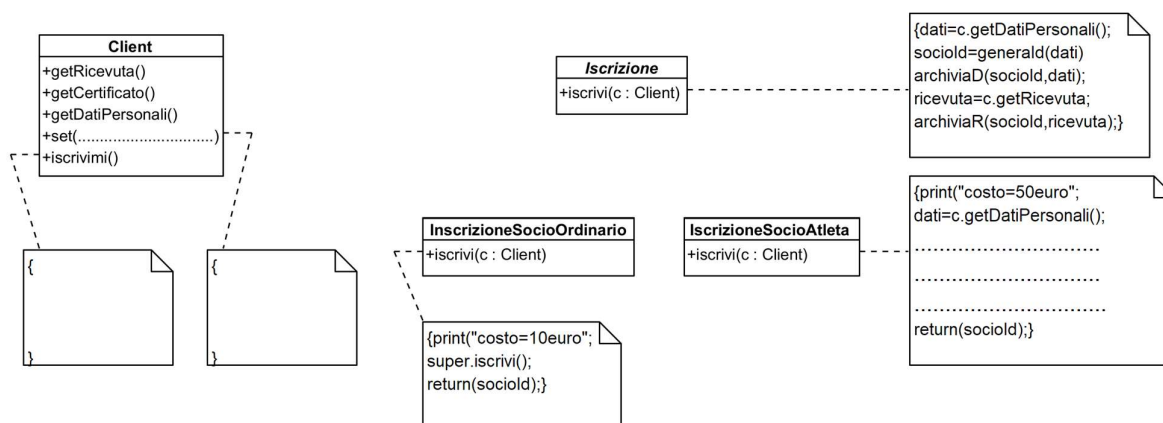
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

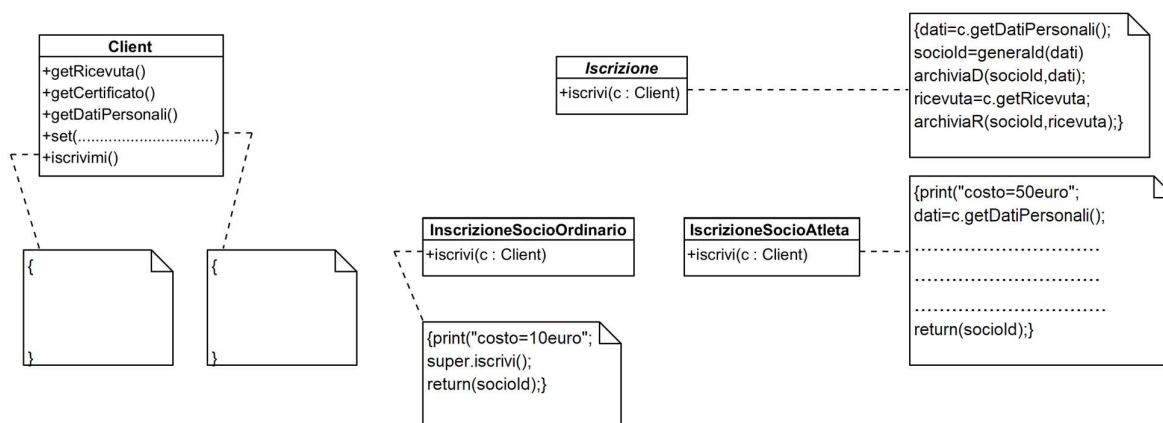
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

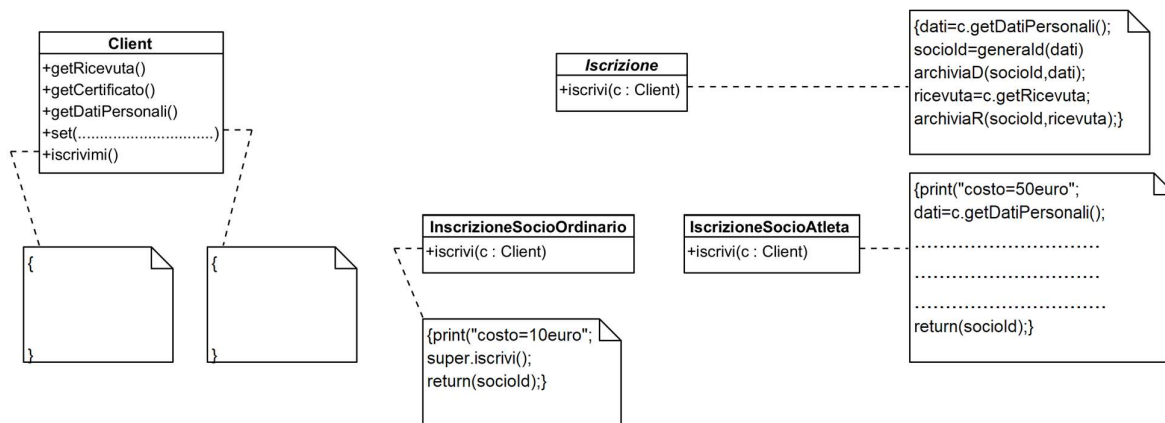
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

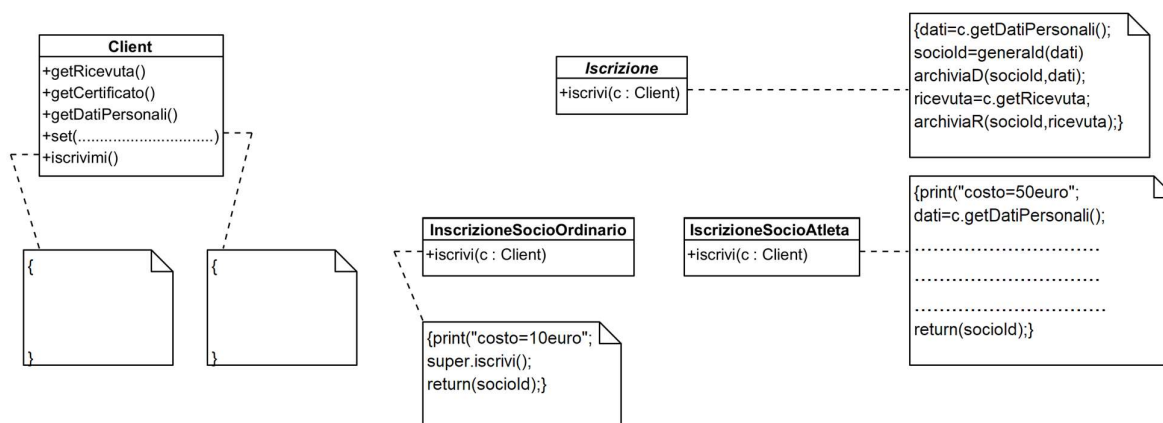
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con “...”

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

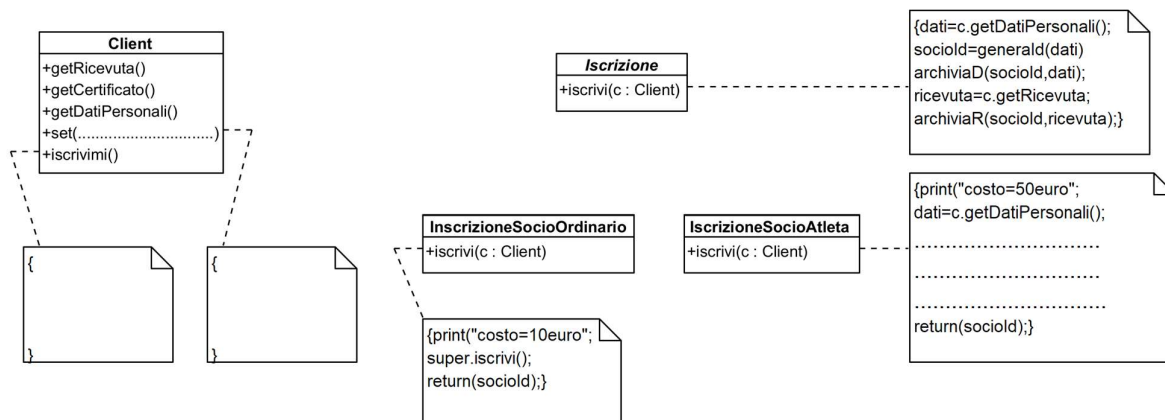
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

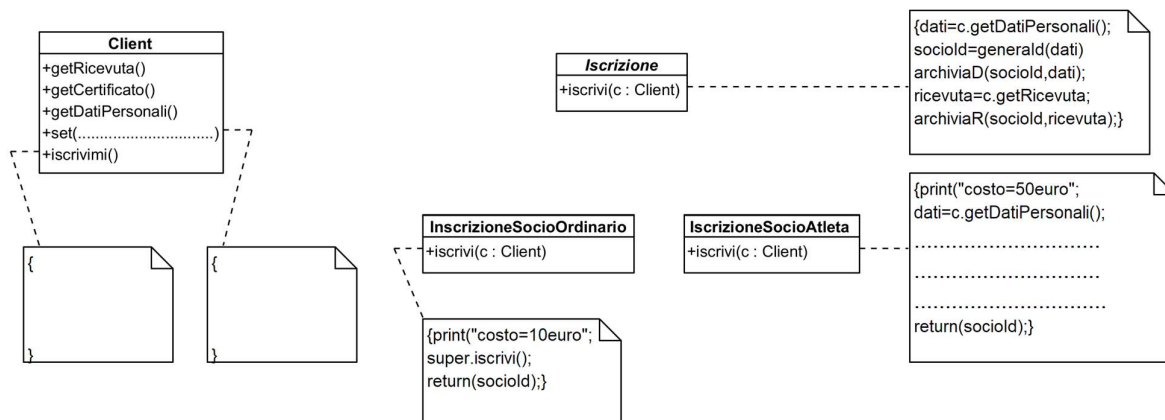
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

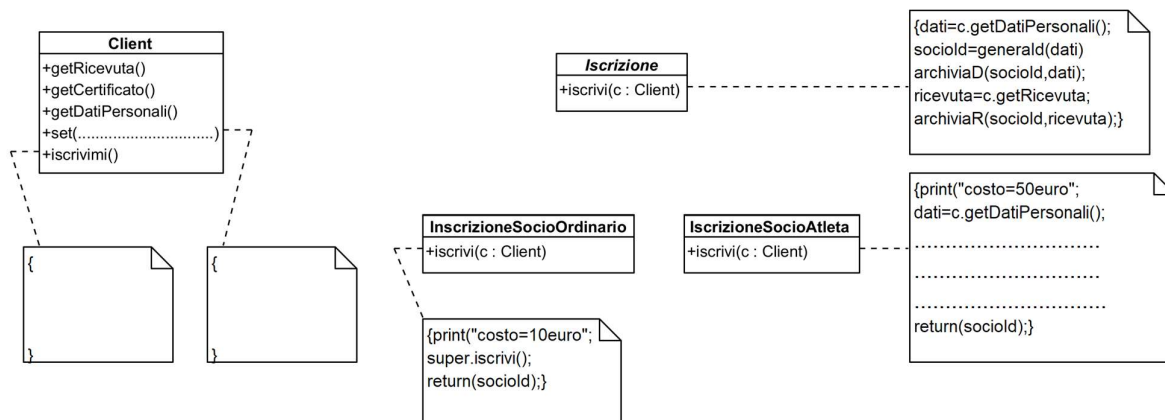
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

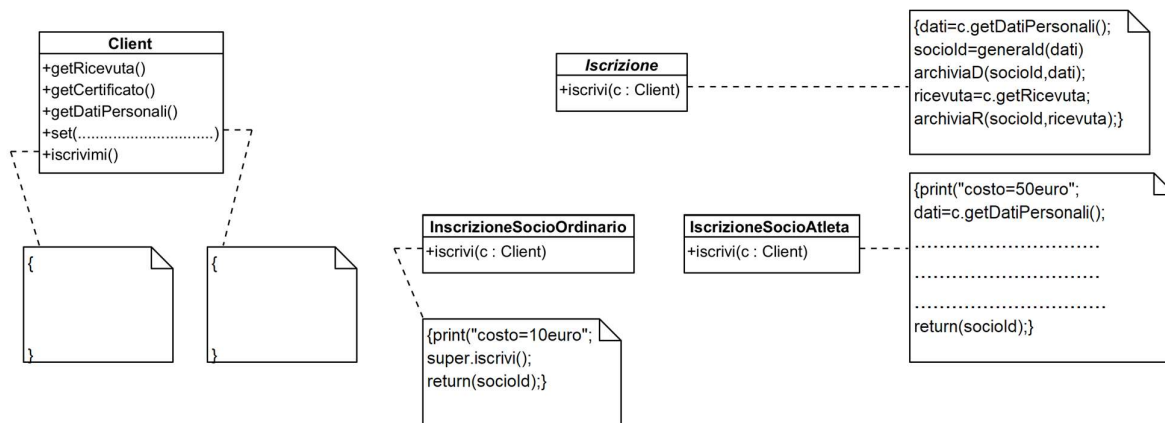
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

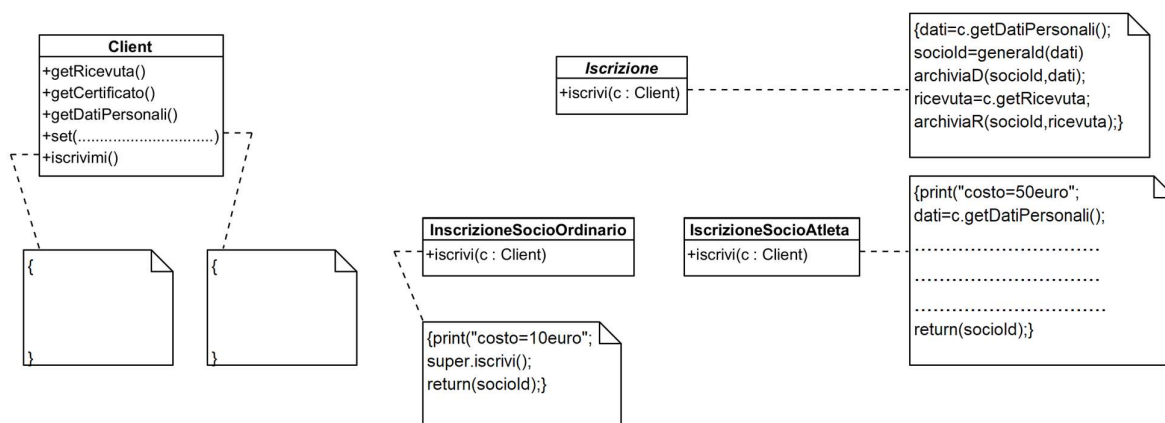
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

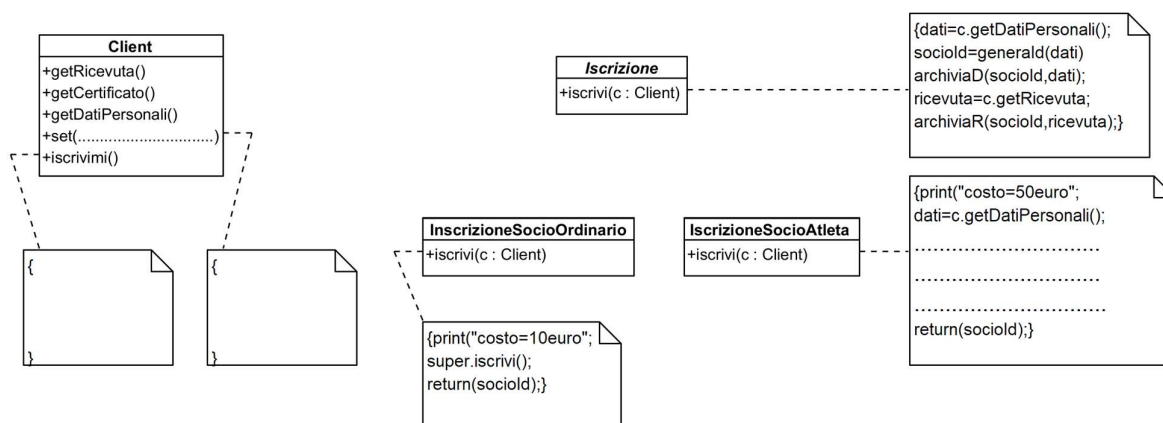
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

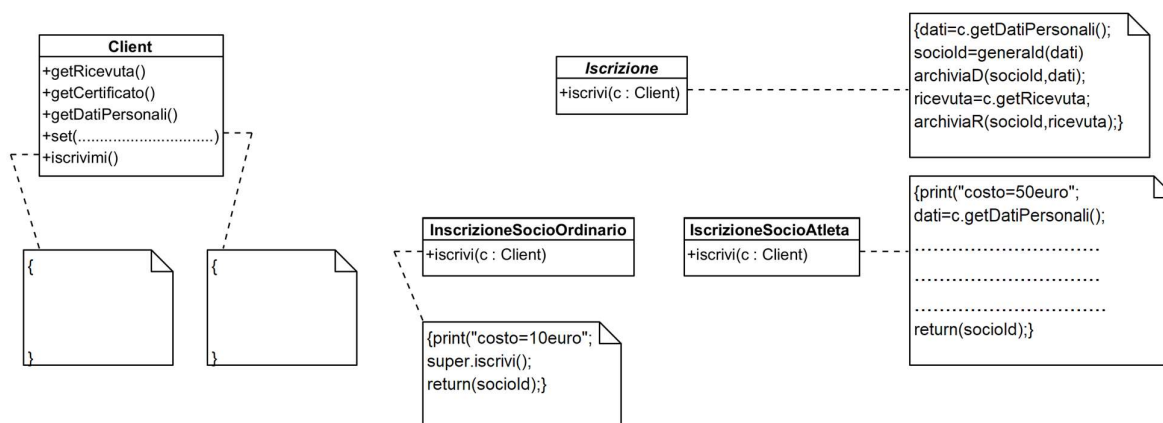
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

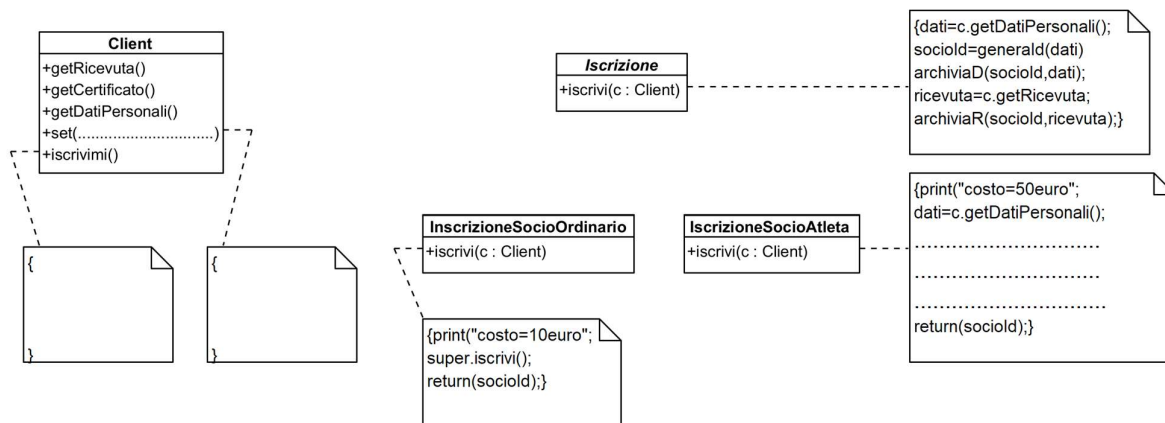
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

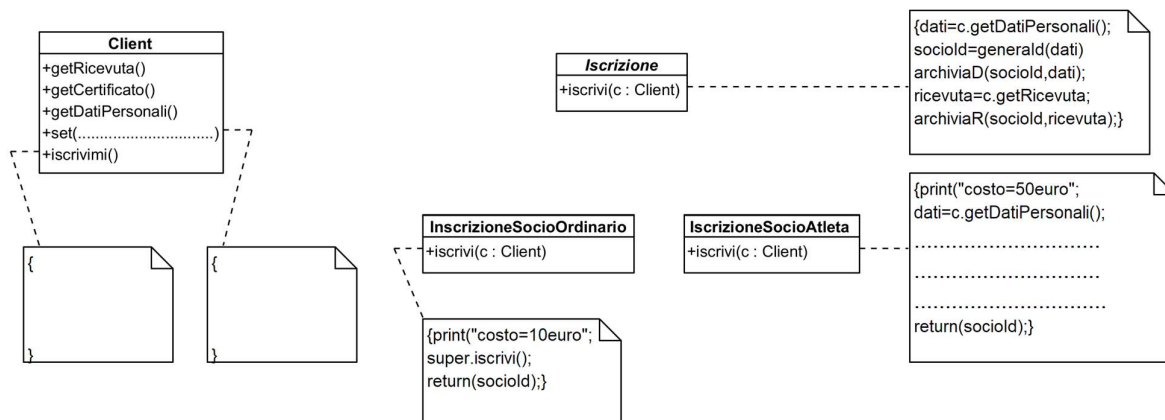
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

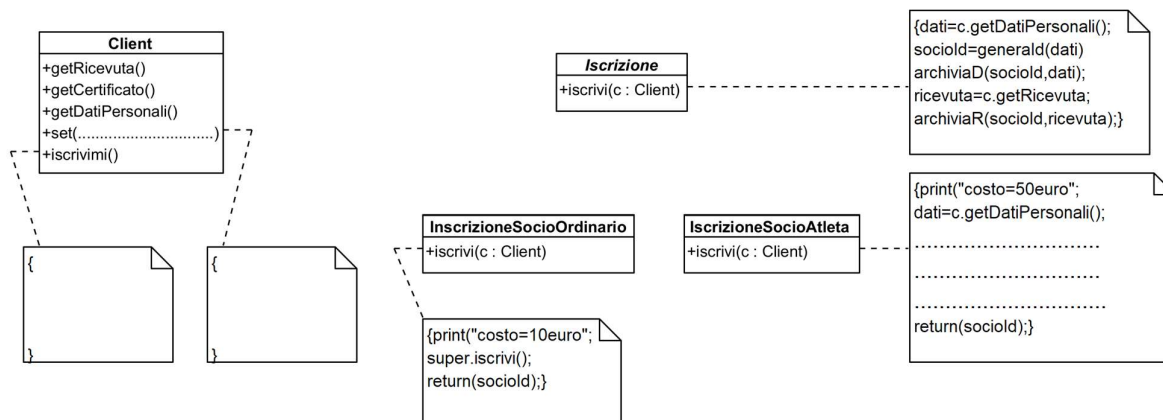
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

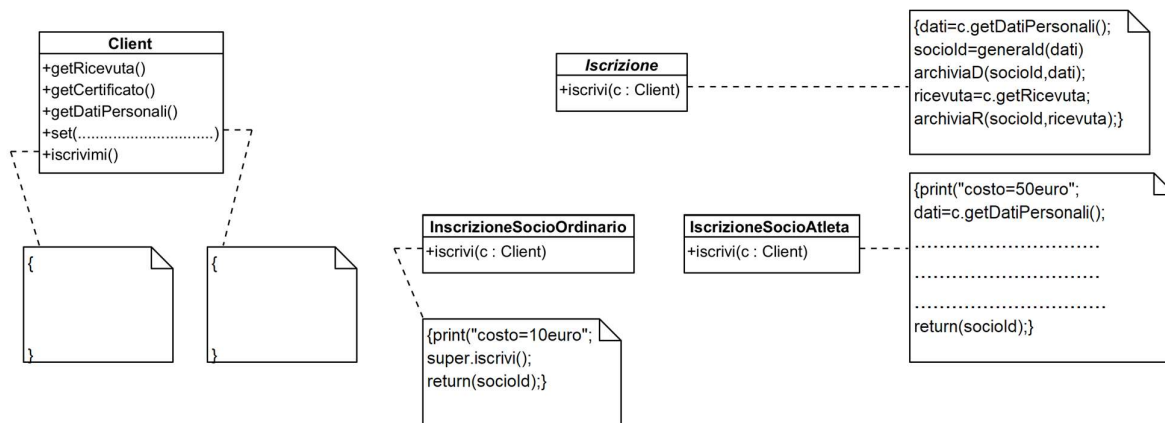
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

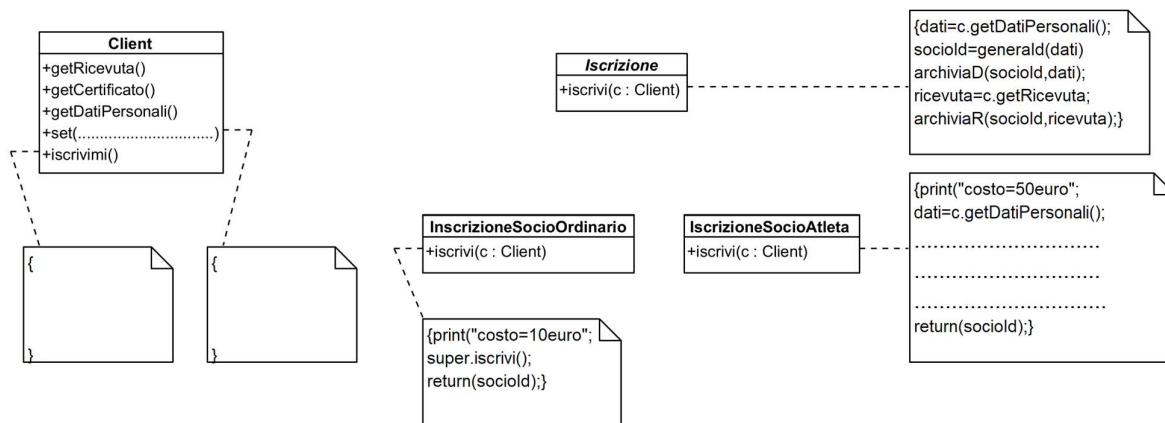
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

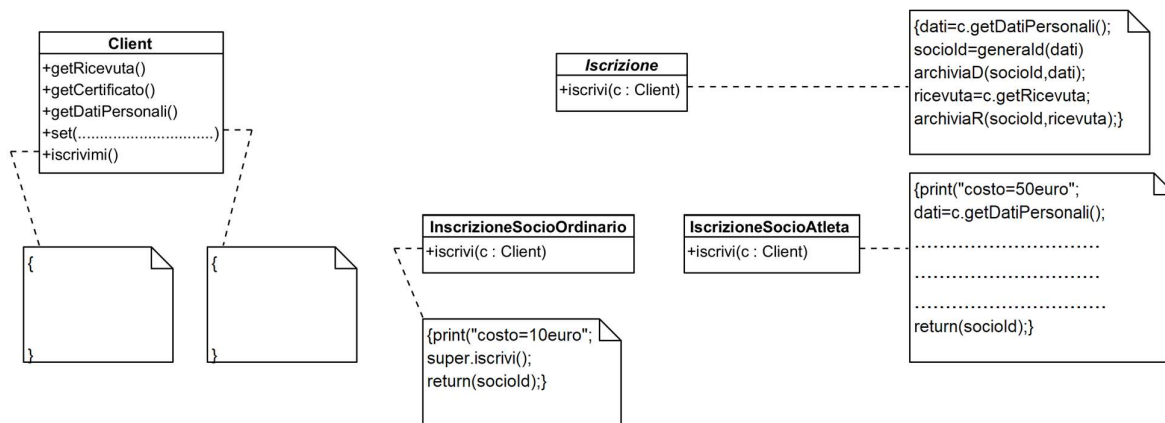
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

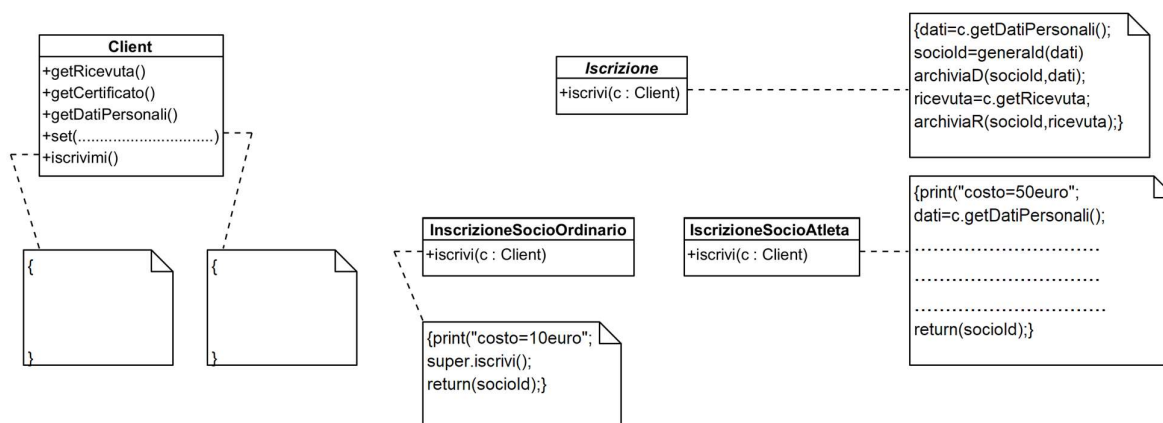
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

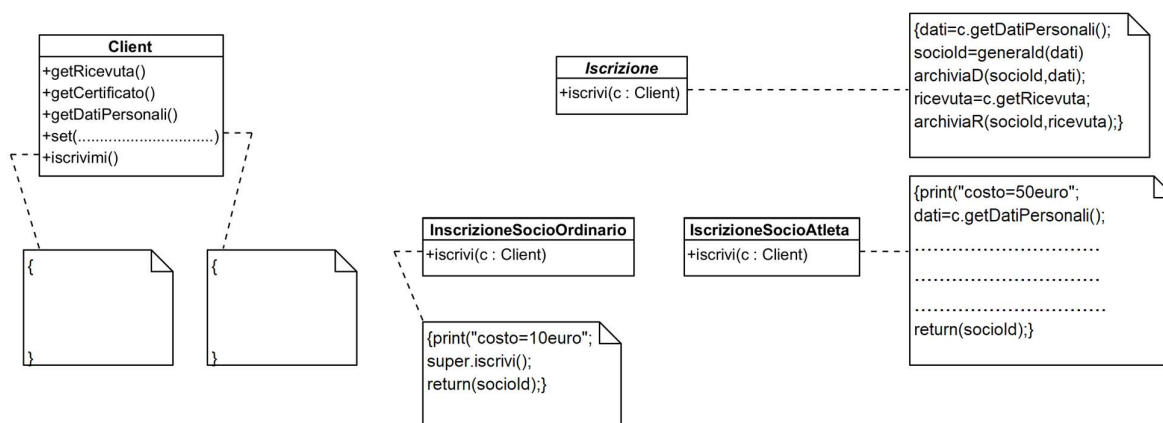
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- a) Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- b) Perché uno stub che risponde in modo random non va bene?

.....

- c) Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

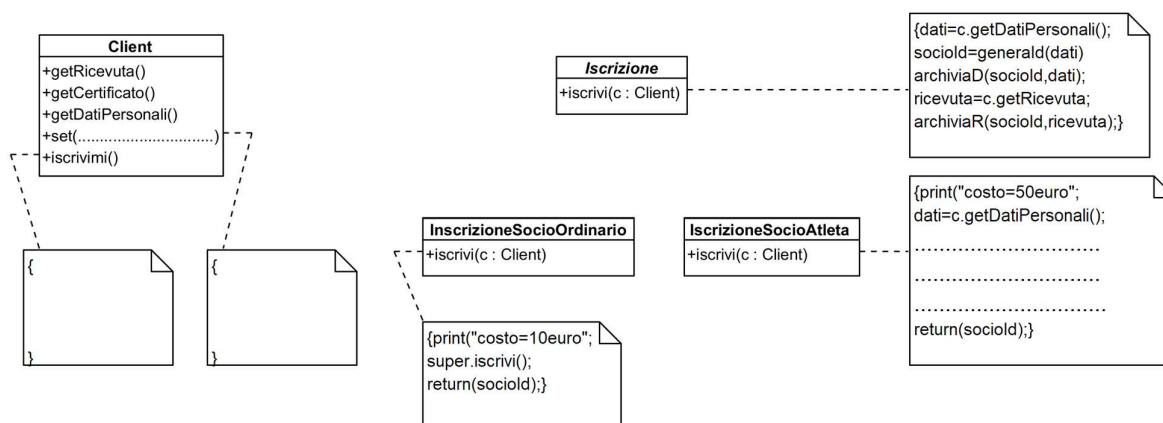
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

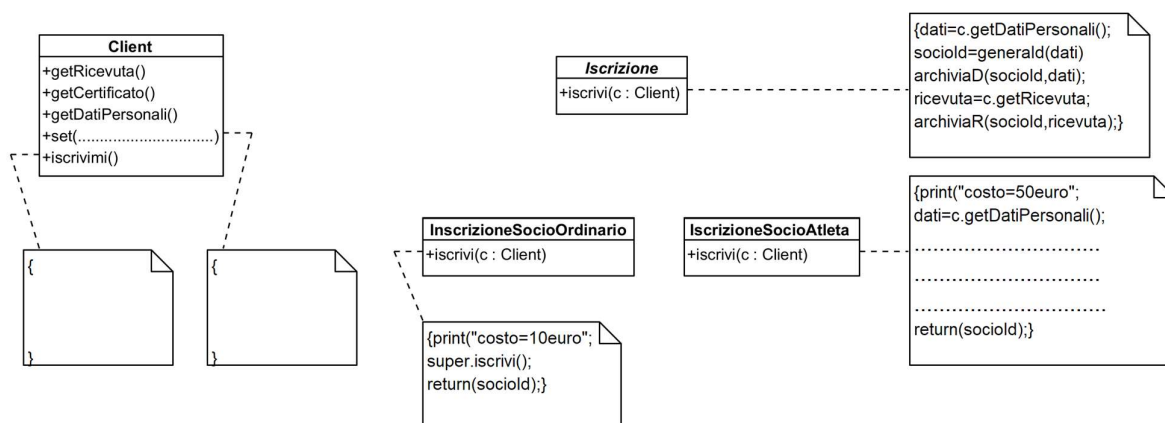
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

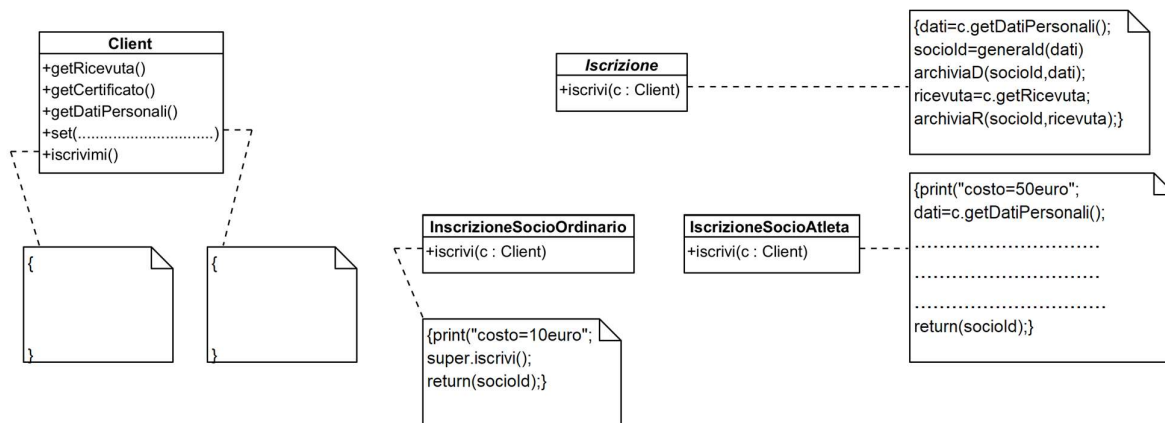
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

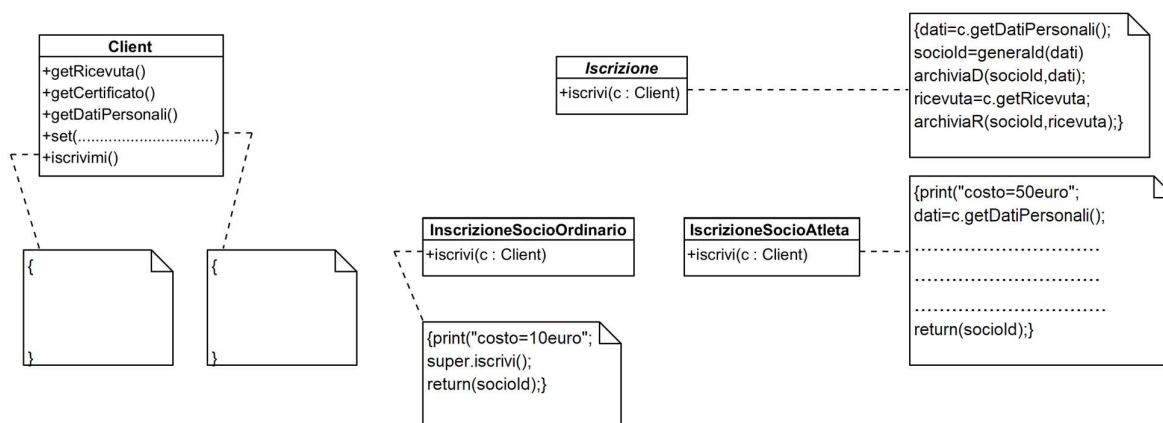
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con "..."

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- Perché uno stub che risponde in modo random non va bene?

.....

- Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....

Sci Club Torre pendente

Lo sci club Torre Pendente accetta soci ordinari e soci atleti. Per associarsi occorre compilare un modulo disponibile sul sito web del club e caricare la ricevuta del bonifico della quota associativa. La segreteria, analizzate le richieste, provvederà a perfezionare le associazioni.

La quota per i soci ordinari è fissata per questa stagione (2023-24) a 10€, permette di partecipare alle attività del club e di avere uno sconto sullo skipass giornaliero degli impianti di Abetone e Cimone. A tal fine, il sistema deve, quando si perfeziona un'associazione, mandare una mail ai gestori degli impianti, comunicando i dati dei soci (nome, cognome, codice fiscale). Il sistema manda inoltre una mail di conferma all'associato.

La quota per i soci atleti per questa stagione è di 50€ e offre gli stessi vantaggi dei soci ordinari. Inoltre, i soci atleti partecipano alle gare del club (una ogni 15 gg durante la stagione) e vengono tesserati alla FISI. Al momento dell'associazione, un socio atleta deve allegare anche un certificato medico con la relativa data di rilascio. Il tesseramento FISI è gestito dalla segreteria, non dal sistema, che però mantiene il numero di tessera, che viene inserito dalla segreteria a tesseramento avvenuto.

Le attività del club prevedono le *domeniche sulla neve*, con trasporto in pullman (GT) tutte le domeniche della stagione (ad esempio dal 21 dicembre al 15 marzo), alternativamente all'Abetone, e al Cimone.

Il costo di una *domenica sulla neve* è fissata per la stagione 23-24 a 15€ a giornata/persona. Le iscrizioni si aprono il lunedì precedente, con modulo e bonifico come per le associazioni. Se si rinuncia a una *domenica sulla neve* entro il giovedì ore 12, si riceve un bonus (in forma di codice di 6 numeri da usare al momento di una successiva l'iscrizione a una *domenica sulla neve*) pari all'intera giornata, altrimenti un bonus di 5€, sempre come codice sconto.

Il sistema deve mantenere tutti i risultati delle gare dei soci atleti. I risultati sono inseriti il lunedì mattina dalla segreteria. I risultati della gara sono visibili a tutti per una settimana, poi solo agli allenatori.

Domanda 1. (Requisiti.) Dare il diagramma dei casi d'uso del sistema che riguardano i soci atleti. Ove non chiaro dal diagramma, dire chi è l'attore principale e per ogni caso d'uso dare pre e post-condizioni.

Domanda 2 (Analisi del dominio) Dare un diagramma delle classi che descrive il dominio del sistema.

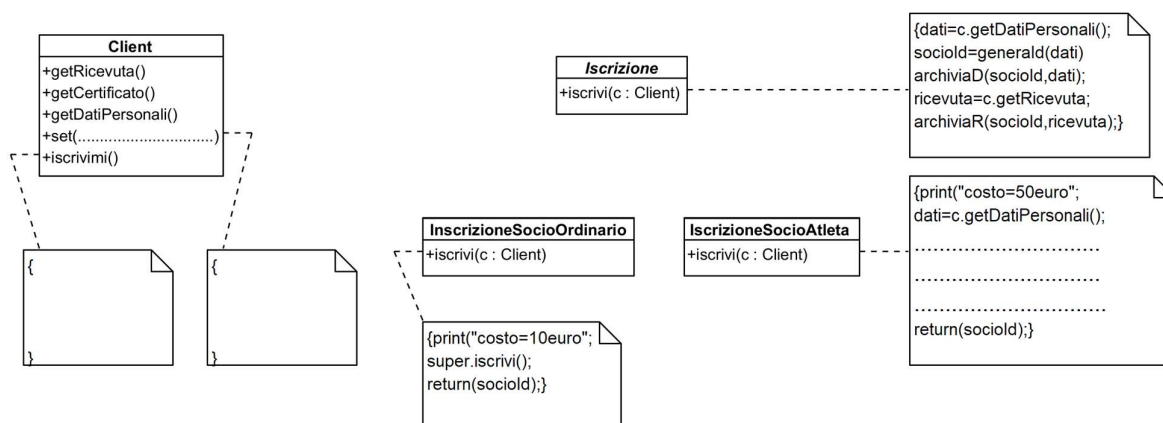
Domanda 3 (Analisi del dominio) Usando un diagramma di macchina a stati, descrivere come evolve un oggetto che mantiene lo stato di un associato, modellando in particolare iscrizioni, cancellazioni, bonus.

Domanda 4 (Analisi del dominio) Usando un diagramma di attività, descrivere il processo di associatura di un socio ordinario. Usare le partizioni per indicare le responsabilità.

Domanda 5 (Progettazione) Dare la vista C&C del sistema, mostrando anche le interfacce offerte dalle componenti.

Domanda 6 (Progettazione) Descrivere, con un diagramma di sequenza, come le component individuate realizzano il caso d'uso "iscrizione a una domenica sulla neve" che ha come attore principale un socio ordinario, nessuna preconditione e come post-condizione "socio iscritto alla domenica sulla neve".

Domanda 7 (completare qui) (Progettazione) Si consideri il seguente diagramma parziale (da completare) che applica un design pattern per implementare i due modi (ordinario e atleta) in cui un cliente può associarsi allo sci club:



a) Di quale design pattern si tratta?

b) Completare il diagramma definendo in particolare le relazioni mancanti e gli spazi lasciati vuoti o con “...”

Domanda 8 (Progettazione) Immaginiamo l'esistenza di un'interfaccia chiamata `IRisultatiGara`, che definisce 2 metodi: per inserire e per visualizzare i risultati di una gara. Successivamente, consideriamo una classe `RisultatiGara` che implementa questa interfaccia senza soddisfare i vincoli di controllo degli accessi. Per affrontare questo problema, intendiamo applicare il pattern Proxy per ottenere una classe `ProxyRisultatiGara`, che sarà responsabile di gestire l'accesso ai risultati delle gare e di controllare i permessi.

- Descrivete questa parte di sistema con un diagramma delle classi
- Scrivete il codice della classe `ProxyRisultatiGara`

Domanda 9 (completare qui) (Verifica.) Il metodo `generateUniqueIntFromString` prende in input una stringa e restituisce un numero intero, serve per generare un codice bonus a partire da una stringa che caratterizza l'associato. Il metodo utilizza una funzione `generateHash` per generare un hash della stringa in ingresso, quindi utilizza un ciclo `while(true)` per verificare se il numero generato è stato generato in precedenza e, in caso affermativo, incrementa il numero di 1. Quando genera un numero unico, lo aggiunge all'insieme dei numeri generati e lo restituisce.

```
import java.util.HashSet;
import java.util.Set;

public class Support {
    private static Set<Integer> generatedNumbers = new HashSet<>();

    public static int generateUniqueIntFromString(String input) {
        int num = generateHash(input);
        while (true) {
            if (!generatedNumbers.contains(num)) {
                generatedNumbers.add(num);
                return num;
            }
            num++;
        }
    }

    private static int generateHash(String input) {
        int hash = 0;
        for (int i = 0; i < input.length(); i++) {
            hash = (hash * 31) + input.charAt(i); // (codice Unicode di charAt(i))
        }
        return Math.abs(hash);
    }
}
```

- a) Scrivere un semplicissimo stub che sostituisca il metodo `generateHash`, restituendo un intero a partire da una stringa, ma non in modo random

```
private static int generateHash(String input) {
    .....
}
```

- b) Perché uno stub che risponde in modo random non va bene?

.....

- c) Dare un insieme minimo di casi di test per avere copertura delle decisioni del metodo `generateUniqueIntFromString`, usando lo stub definito al punto a.

.....

.....

.....

.....