

- Scrivere immediatamente, su ogni foglio che vi è stato consegnato, cognome, nome, numero di matricola.
- Non è consentito consultare appunti, libri, colleghi, né qualunque dispositivo elettronico, pena l'immediato annullamento della prova.
- Tempo a disposizione: 3 ore.

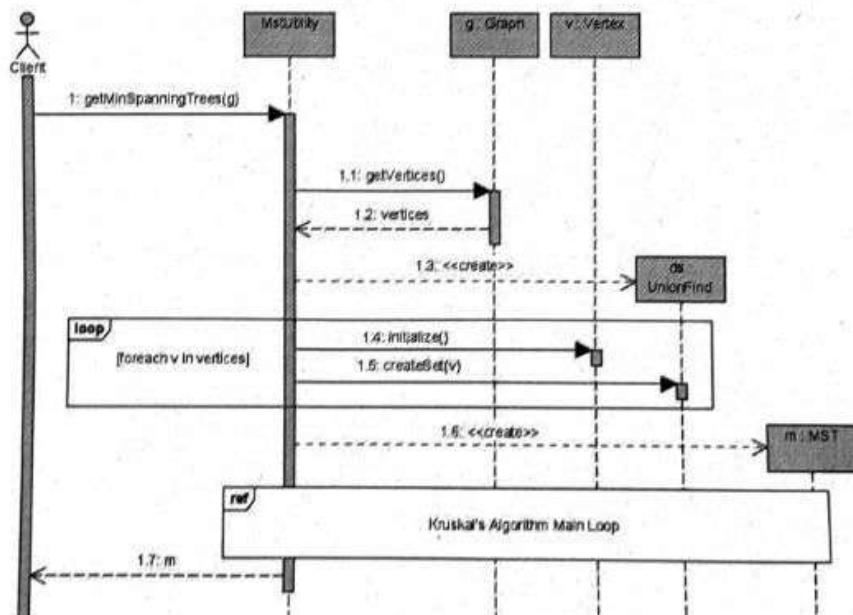
### Esercizio 1

Si vuole progettare un software di supporto alla gestione di una centro culturale comunale non-profit e self-service. Il sistema permette a un amministratore di gestire i libri/CD musicali/DVD presenti in inventario, e ai cittadini di prendere in prestito libri/CD/DVD, interagendo con i totem touch-screen presenti in sede. In particolare, previa autenticazione con Carta di Identità Elettronica, un cittadino può prendere in prestito un oggetto selezionandolo tra quelli correntemente disponibili in inventario. Opzionalmente, l'utente può anche filtrare gli oggetti disponibili effettuando ricerche per titolo oppure per genere. Dopo aver selezionato l'oggetto di interesse, l'utente indica una data di termine del prestito (che non può eccedere i 30 giorni), entro la quale si impegna a rendere l'oggetto, e conferma il prestito. Un utente non può prendere in prestito più di cinque oggetti alla volta. In seguito alla conferma, il sistema utilizza API messe a disposizione dal componente robotizzato "InventoryRetriever", sviluppato da una terza parte, per recuperare l'oggetto di interesse dal magazzino e consegnarlo automaticamente in un apposito vano predisposto in sede, dove l'utente potrà ritirarlo.

- Dettagliare il caso d'uso relativo alla funzionalità di presa in prestito di un oggetto per mezzo di mock-up e descrizioni testuali strutturate secondo il formalismo di Cockburn. Usare la propria conoscenza del dominio per derivare dettagli non definiti nei requisiti.
- Definire un class diagram di analisi del sistema, inteso come modello di dominio, relativo al caso d'uso della presa in prestito di un oggetto. È possibile rifarsi alle euristiche EBC, e di Abbott.
- Fornire un sequence diagram di analisi per il caso d'uso relativo alla presa in prestito di un oggetto.

### Esercizio 2

Si scriva tutto il codice Java che è possibile desumere dal seguente sequence diagram.



### Esercizio 3

Il metodo `getRentalPrice` della classe `Library` viene utilizzato per calcolare il costo del noleggio di un libro. Il metodo prende in input tre parametri:

- `int year` è numero intero non negativo rappresentante il numero di anni dalla prima uscita del libro.
- `int weeks` è un intero compreso tra 1 e 52, rappresentante la durata in settimane del prestito.
- `String profile` è una stringa rappresentante il livello del cliente, e può assumere valori in {`bronze`, `silver`, `gold`, `platinum`}

Il metodo `getRentalPrice`, se i parametri in input sono validi, ritorna un costo (in €) calcolato moltiplicando il numero di settimane per un prezzo settimanale dipendente dal livello del cliente e dal numero di anni trascorsi dall'uscita del libro, determinato secondo la tabella seguente.

LIVELLO	ANNI TRASCORSI DALLA PRIMA USCITA				
	0	[1, 2]	[3,4]	[5,9]	10+
Bronze	5.00 €	4.00 €	3.00 €	2.00 €	1.50 €
Silver	4.50 €	3.50 €	2.50 €	1.50 €	1.00 €
Gold	4.00 €	3.00 €	2.00 €	1.00 €	0.75 €
Platinum	3.50 €	2.00 €	1.00 €	0.75 €	0.50 €

Se uno o più parametri non sono validi, il metodo solleva una `IllegalBookRentalException`.

- Indicare le classi di equivalenza individuate.
- Scrivere quattro test JUnit con strategia Black Box per il metodo `getPrice`, indicando per ciascuno di essi quali classi di equivalenza copre. Si richiede inoltre che un test corrisponda a scenari in cui i parametri non sono validi, e che i restanti tre corrispondano a scenari in cui i parametri sono validi.
- Quanti test sono necessari per testare il metodo con strategia WECT? Quanti con SECT? Motivare la risposta.

### Esercizio 4

Una stampante, una volta accesa, rimane in attesa dell'invio di documenti da stampare. In presenza di un documento da stampare, la stampante rifiuta tutte le successive richieste di stampa finché non ha ultimato la stampa corrente. In ogni momento, in presenza di un apposito segnale inviato dal produttore della stampante via Internet, la stampante può interrompere le sue operazioni correnti per scaricare e poi installare un aggiornamento. Al termine dell'installazione dell'aggiornamento, le attività eventualmente interrotte riprendono.

Si rappresenti il comportamento della stampante sopra descritta utilizzando il formalismo degli StateChart. Si richiede esplicitamente che la modellazione sia gerarchica e che siano utilizzati history state.