UNIVERSIDAD AUSTRAL

Facultad de Ingeniería

Trabajo Práctico Cuatrimestral: Especificación

Materia:

- Algoritmos y Estructuras de Datos

Profesores:

- Ignacio Cassol
- Alicia Gioia

Alumnos :

- Lucas Manzanelli
- Santiago Hazaña

Exercise 5: MovieClub

a. ADT: MovieClub

<u>*Description:*</u> MovieClub has three lists which represents Clients, Movies and Excesses.

i. Constructors:

- ***//createMovieClub: - →*** **MovieClub**
    - Precondition: -
    - Postcondition: Initialize movieclub struct

ii. Destructors:

- ***freeMovieClub: MovieClub → -***
    - *Free movieclub where it's allocated in the memory.*
    - Precondition: MovieClub initialize.
    - Postcondition: MovieClub destroy.

b. ADT: Client

Description: Client who can rent a Movie from MovieClub and has MovieCard.

i. Constructors:

- ***createClient: String: name X String: surname X int:DNI →*** **Client**
    - Precondition: unique DNI.
    - Postcondition: Client initialize.

ii. Modifiers:

- ***pickUpMovies: StaticList: Movies → -***
    - *Takes movies from the list, comparing client's DNI with movie's moviecardID.*
    - Precondition: Unique client's DNI.
    - Postcondition: Movies add in clients.
- ***returnMovie: MovieClub X String: movieTitle → -***
    - *Selects a Movie from client with movieTitle and returns it to movieclub.*
    - Precondition: Unique client's DNI.
    - Postcondition: Movie remove in client.

iii. Destructors:

- ***freeClient: Client → -***
    - *Free client where it's allocated in the memory.*

- Precondition: Client initialize.
- Postcondition: Client destroyed.

c. ADT: Excess

Description: A special report with all information necessary to show from movieClub.

i. Constructors:

- ***createExcess: - → Excess***
  - Precondition: -
  - Postcondition: -

ii. Modifiers:

- ***determinateUnreturnsMovies: Movieclub→ -***
  - *Calculates the un-returns movies from movieClub.*
  - Precondition: MovieClub initialize.
  - Postcondition: modifies un-return movies quantity in Excess.
- ***registerIncome: Movieclub→ -***
  - *Calculates the income from movies.*
  - Precondition: MovieClub initialize.
  - Postcondition: modifies income quantity in Excess.
- ***rewardClients: Movieclub→ -***
  - *Every three movies, reward the client with five amount in the movieCard.*
  - Precondition: Client has a movieCard.
  - Postcondition: increase MovieCard’s amount.

iii. Analyzers:

- ***determinateExcess: Excess→ -***
  - *Show excess’s information.*
  - Precondition: All modifiers must be initialized before.
  - Postcondition:-

iv. Destructors:

- ***freeExcess: Excess → -***
  - *Free excess where it’s allocated in the memory.*
  - Precondition: Excess initialize.
  - Postcondition: Excess destroyed.

d. ADT: Manager

Description: The person who organized the Clients and Movies in Movieclub.

i. Constructors:

- ***createManager: String: name X String: surname X int: DNI → Manager***

- Precondition: -
- Postcondition: Manager initialized.

ii. **Modifiers:**

- ***registerClient: Movieclub X Client→ -***
    - *Add client into Movie Club.*
    - Precondition: MovieClub initialize.
    - Postcondition: Clients now it's allocated in movieClub
- ***generateMovieCard: Client X double: amount → -***
    - *Calculates the income from movies.*
    - Precondition: MovieClub initialize.
    - Postcondition: modifies income quantity in Excess.
- ***generateExcess: Movieclub→ -***
    - *Create an excess with all modifiers method initialize, by using movieclub information.*
    - Precondition: MovieClub initialize.
    - Postcondition: add Excess in MovieClub.
- ***increaseAmountMovieCard: MovieCard X double: Amount→ -***
    - Precondition: MovieCard initialize.
    - Postcondition: increase MovieCard's amount
- ***saveMovie: MovieClub X MovieCard X Movie→ -***
    - Recibes the movie and movieCard for returns.
    - Precondition: MovieClub, MovieCard and Movie initialized.
    - Postcondition: it's necessary decrease MovieCard's amount.

iii. **Destructors:**

- ***freeManager: Manager→ -***
    - *Free manager where it's allocated in the memory.*
    - Precondition: Manager initialize.
    - Postcondition: Manager destroyed.

e. ADT: MovieCard

Description: A client's card to rent movies and checks specific movies .

i. **Constructors**:

- ***createMovieCard: int: id X double: amount→ MovieCard***
    - Precondition: id must coincide with client's DNI.
    - Postcondition: MovieCard initialize

ii. **Modifiers:**

- ***checkAvailableMovie: StaticList: Movies X String: movieTitle→ boolean***

    - *Find the movie by this title.*
    - Precondition: Movies initialize.
    - Postcondition: returns true if movie was found and anyone has already rented.
- ***checkMoviePremiere: StaticList: movies → -***
    - *Show all today premieres movies.*
    - Precondition: -
    - Postcondition: -

iii. **Analyzers:**

- ***showMovies: StaticList: Movies→ -***
    - *Show movies title and price .*
    - Precondition: -
    - Postcondition:-

iv. **Destructors:**

- ***freeMovieCard: MovieCard→ -***
    - *Free moviecard where it's allocated in the memory.*
    - Precondition: Moviecard initialize.
    - Postcondition: Moviecard destroyed

f. ADT: Movie

Description: Represents the rental with a Movie.

i. **Constructors**:

- ***createMovie: String: title X double: amount → Movie***
    - Precondition: -
    - Postcondition: Movie initialize

ii. **Modifiers:**

- ***changePrice: Movie X double: amount→ -***
    - Precondition: Movie initialize.
    - Postcondition: modifies Movie's price.
- ***rentMovie: Movie X int: idMovieCard X double: amount → -***
    - *Rent the Movie by changing the id with idMovieCard.*
    - Precondition: Movie's id must be zero.
    - Postcondition: modifies Movie's id.

iii. **Analyzers:**

- ***isAvailable: Movie → -***
    - *Show if Movie is available for rent.*
    - Precondition: -
    - Postcondition:-

iv. **Destructors:**

- ***freeMovie: Movie → -***

- *Free movie where it’s allocated in the memory.*
- Precondition: Movie initialize.