

ACTIVITAT 7.3 Taller d'ITV

Temporalització: Sense determinar

Agrupament: INDIVIDUAL

Què necessites per a fer esta activitat?

- ✓ Entorn de desenvolupament Netbeans
- ✓ Aplicar la metodologia de desenvolupament top-down orientada a objectes





Enunciat

L'objectiu d'esta pràctica és realitzar un programa Java que siga capaç de gestionar la següent especificació:

Un **taller** de revisió d'ITV vol gestionar els **vehicles** que van rebent per a la revisió així com les cues que estos vehicles van provocant als diferents boxs del taller on són atesos.



El taller de revisió es compon de **6 boxs**, tots ells amb les mateixes responsabilitats de revisió de vehicles, és a dir, indistintament del box pel qual un vehicle passe, passarà per les mateixes fases de revisió, que són sempre 4 i amb el següent ordre:

1. Revisió inicial d'elements de **seguretat** 
2. Revisió del **sistema elèctric** 
3. Revisió d'**emissió de fums** 
4. Revisió de **frens** i direcció. 

Des del sistema es tindrà la capacitat decidir en qualsevol moment que els vehicles que estiguen dins un box es moguen tots alhora, de la fase de revisió on estiguen cap a la fase següent. S'assumeix en este cas que estan tots llestos per a passar de fase i que un suposat mecànic els ha donat el vistiplau.

Quan un vehicle arriba al taller se li pren tant el número de matrícula (4 números seguits de 3 lletres) com el model i del vehicle, a més del tipus que pot ser (cotxe, microbús, furgoneta o camió). Si el registre del vehicle és correcte (no hi pot haver dos vehicles amb la mateixa matrícula), este vehicle es posarà automàticament a la cua general.

La cua general no pertany a cap box concret, podent passar a la cua de fases de revisió del box que des del programa informàtic es decidisca. Lògicament, la primera fase de revisió del box que reclame al següent vehicle de la cua general ha d'estar lliure per atendre este vehicle, sinó se li denegarà eixe reclam.

Les **opcions** que el programa ha de presentar al **menú** són les següents:

1. **Alta i recepció de vehicles:** s'encarregarà de demanar les dades de registre del vehicle i posar-lo automàticament a la cua general. No es podran introduir a la cua dos vehicles amb la mateixa matrícula. Si alguna de les dades s'introdueix incorrectament, es tornarà a demanar la dada fins que s'introduïska bé. Cal informar l'usuari sobre en quina posició de la cua general es troba el vehicle rebut.
2. **Reclamar vehicle per a entrar al box:** es preguntarà pel box que desitja rebre el següent vehicle que es troba a la cua inicial, si és que n'hi ha algun i hi ha lloc per a ell en eixe box (no hi ha d'haver cap vehicle en la primera fase de revisió d'eixe box). Si no s'introdueix un número de box correcte, es tornarà a demanar fins que siga vàlid. Si no és possible l'entrada del vehicle al box, es denegarà esta operació informant a l'usuari (tant en el cas favorable com al desfavorable de l'operació).
3. **Moure tots els vehicles de fase dins d'un box.** Es preguntarà pel número de box on es vol desplaçar una fase a tots els vehicles dins seu. Per descomptat, si un vehicle es troba a l'última fase, quedarà ja fora del box.
4. **Informació de l'estat d'un box concret:** es preguntarà pel número de box (entre 1 i 6). Es mostrarà en quin estat es troba cada fase del box, mostrant de cadascuna quin vehicle té assignat (mostrant la seua matrícula, marca, model i el seu tipus), si el té.
5. **Informació general de tots els boxes:** similar a l'opció anterior però pels 6 boxes del taller.
6. **Eixir del programa.**

Mètode de treball

1. Dissenya el projecte en diferents classes utilitzant la metodologia top-down. Pots començar per exemple per la classe Taller. Recorda que en esta classe és on ha d'estar el mètode main i que disposarà d'un mètode "inici" no estàtic que donarà lloc a l'inici del programa. No oblidis implementar el constructor de la classe.
2. El començament del programa mostrarà el menú d'opcions. En funció de l'opció seleccionada cal fer una acció o una altra. Pots seguir l'estratègia de menú d'opcions utilitzada en altres programes.
3. Observa que el programa ha de treballar amb una gestió de cues (el primer en entrar, ha de ser el primer en eixir). No hi ha una única solució per a esta gestió.
4. Has d'informar l'usuari de totes les operacions que es van fent conforme es van executant les diferents opcions, tant les favorables com les desfavorables i seguint l'ordre establert a l'especificació del programa.
5. En cas d'introduir una dada que no siga vàlida, per exemple, per no complir amb un cert format, has de tornar-les sempre a demanar fins que s'introduïsquen correctament. No confongues això amb que una opció determinada del menú no acabe sent favorable. No obstant això, no cal que valides els tipus de dades de les dades demanades (per exemple, quan utilitzàvem els mètodes hasNextXXX de Scanner). Utilitza la classe GestorIO per demanar i mostrar informació.
6. Exporta el projecte en un fitxer zip i puja'l a la tasca corresponent, encara que és altament recomanable l'ús de GitHub. Així, tindràs control en tot moment dels canvis que aniràs fent.
7. És molt recomanable que vages realitzant tests a les classes que vas implementant, bé de manera local a cada classe, o bé en un paquet extern de tests.
8. Pots utilitzar sentència "assert" quan vulgues preconditionar la invocació a un mètode, però recorda que només s'utilitza a la fase d'implementació i no al desplegament de l'aplicació.