

# PROJECTE D'UN SOFTWARE PER A UN MAGATZEM DE PECES ELECTRÒNIQUES

**Autores:**

Marina Navarro Sato

Núria Martín Xifré

**Grup de Laboratori: 2**

**Professor:**

Fatos Xhafa Xhafa

# ÍNDEX

Introducció	3
Objectius	3
Requisits	3
Disseny i decisions d'implementació	4
Proves i posta a punt	10
Conclusions	10
Autoavaluació	11
Bibliografia	11

## 1.INTRODUCCIÓ

El projecte realitzat té com a finalitat la creació d'un programa informàtic eficient que gestioni els productes continguts en un magatzem d'un negoci de peces electròniques. En aquesta memòria s'exposen els objectius que ha de complir el software, una guia en línies generals de l'estructura i el contingut del programa i el testeig pel qual ha passat, així com unes conclusions i pensaments finals.

## 2.OBJECTIUS

Els objectius plantejats en la realització del projecte són els següents:

- Implementar un programa informàtic que ajudi a solucionar un problema real.
- Realitzar que resolgui tots els problemes plantejats i ho faci de manera eficient.
- Aprendre nous conceptes, com la generació de codis EAN 13, la llibreria time.h i la programació amb fitxers.
- Realitzar una memòria clara, concisa i que resolgui qualsevol dubte que el programa pugui generar sobre el seu funcionament.
- Resoldre tots els dubtes que el programa pugui plantejar en els punts oberts d'aquests i aplicar el criteri que pensem que sigui més convenient.
- Intentar millorar el programa i fer-lo interactiu per a l'usuari.

## 3.REQUISITS DEL PROJECTE

El programa de gestió del magatzem de peces electròniques ha de constar d'una sèrie de requisits essencials per al seu bon funcionament. El llistat de funcionalitats és el següent:

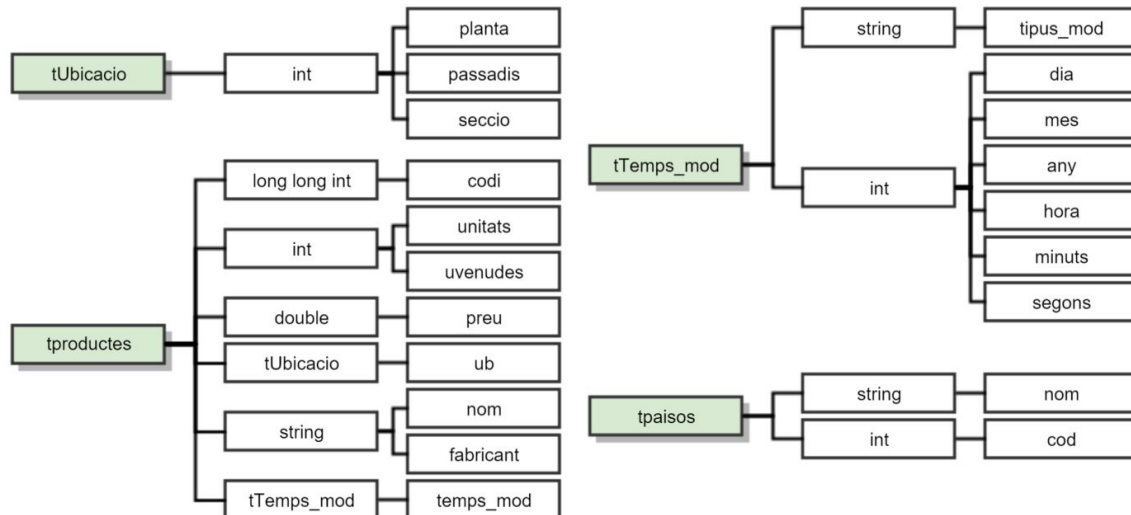
- Donar d'alta, de baixa i modificar un producte. Per a donar d'alta un producte és necessària la generació d'un codi EAN 13 per a cada producte. (Bloc I)
- Cercar productes segons el seu codi, nom i fabricant. (Bloc II)
- Llistar els productes amb un stock inferior a un mínim marcat. (Bloc III)
- Simular la venda d'un producte. (Bloc III)
- Simular la comanda de compra de productes. (Bloc III)
- Llistar els productes segons el nombre de vendes de cadascun en ordre descendent. (Bloc III)
- Recuperar i guardar dades de fitxers. (Bloc IV)
- Utilitzar ordenació i cerca binària.
- Guardar l'hora del sistema en què un producte ha sigut donat d'alta o modificat.

Els requisits anteriors són els demanats en l'enunciat del projecte. Ara bé, les funcionalitats del programa es poden ampliar per a fer-lo més interactiu per a l'usuari.

## 4.DISSENY I DECISIONS D'IMPLEMENTACIÓ

Els esquemes que es proposen en aquest apartat són simplifications amb la finalitat d'aclarir la forma del programa, per la qual cosa no són gaire precisos. Les accions i funcions no descrites es considera que són prou senzilles com per no necessitar explicació.

Hem decidit crear les següents **tuples** per tal d'agrupar la informació de manera eficient:

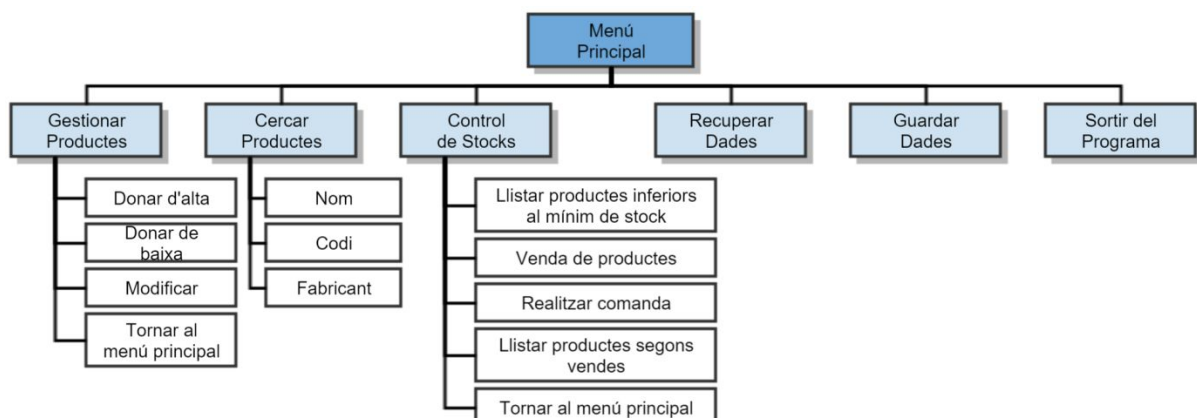


Les dades dels productes són guardades en el vector “producte” de tipus “tproductes” que, a mesura que se’n van afegint, s’ordenen segons el codi de barres que li hagi sigut assignat amb la funció “ordenar\_insercio”. Això permet fer cerques binàries segons el codi de barres, que és únic.

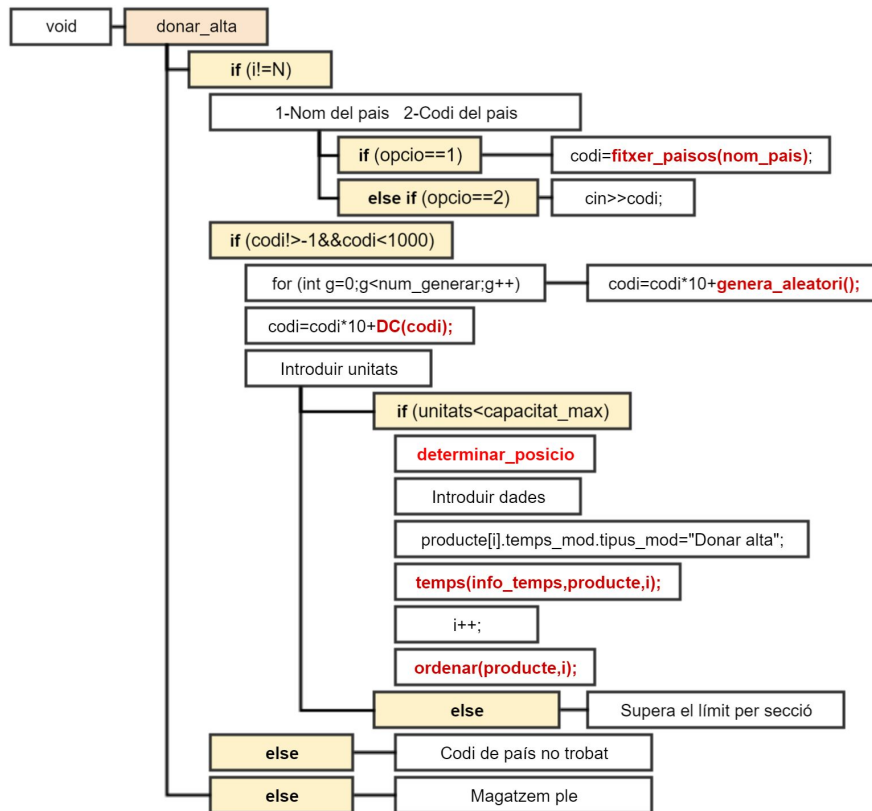
A dins de “tproductes” s’hi troba “ub” (ubicació): a mesura que es s’afegeixen articles a la base de dades se’ls assigna, per defecte, la posició al magatzem consecutiva a la de l’últim introduït. Així doncs, a una mateixa secció en certa planta i passadís s’hi troba un únic tipus de producte, tal i com succeeix als magatzems de l’Ikea. Cal esmentar que hi ha un vector de tipus “tUbicacio” que guarda les seccions buides causades per l’eliminació d’algun producte. Aquests espais buits tenen prioritat a omplir-se quan es mouen o s’afegeixen articles ja que les plantes baixes, els primers passadissos de menor número i les primeres seccions considerem que són les que queden més a prop de l’entrada.

“temps\_mod” (temps de darrera modificació): es guarda la data i l’hora de la darrera modificació de la seva informació, “donar d’alta” o bé “modificació” de les dades.

El **menú** principal del programa i els **submenús** destacables s’estructuren de la manera següent:



L'acció “**donar\_alta**” primer comprova que el magatzem no estigui ple. Després se li assigna el codi: el 3 dígits corresponents al país poden introduir-se directament o bé buscar-los en un

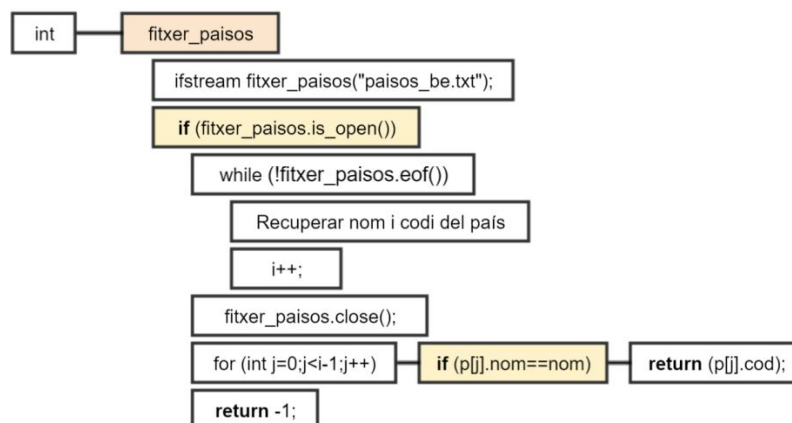


fitxer.

Comprova que el codi sigui vàlid i llavors es demana la quantitat d'unitats i es comprova que no sobrepassi el límit de capacitat de les seccions.

A continuació s'acaben de determinar les dades, es guarda el temps en què s'ha modificat, s'augmenta en 1 el nombre de productes que hi ha al magatzem i s'ordena el vector de productes segons el codi.

La funció “**fitxer\_paisos**” determina quin codi té el país especificat. Així doncs, recupera de l'arxiu “paisos\_be.txt” els noms i codis dels països i després busca el nom que coincideixi amb el que l'usuari ha introduït i retorna el seu codi. En cas que no el trobes retorna -1, un valor sentinella ja que cap codi és negatiu. “Recuperar nom i codi” es veu amb més exemples en l'organigrama de “recuperar\_dades”.

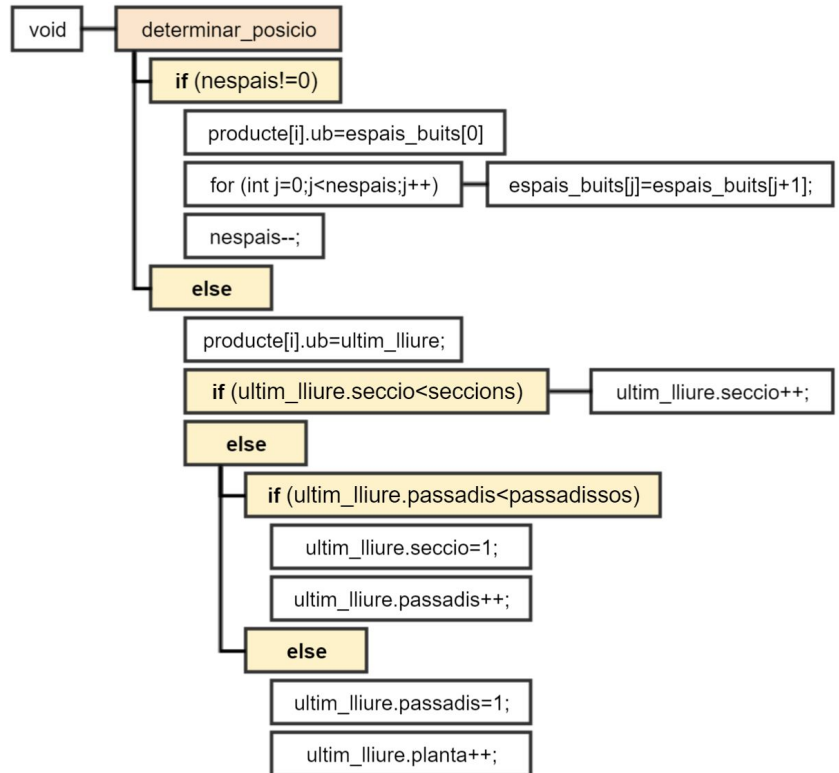


L'acció “**determinar\_posicio**”

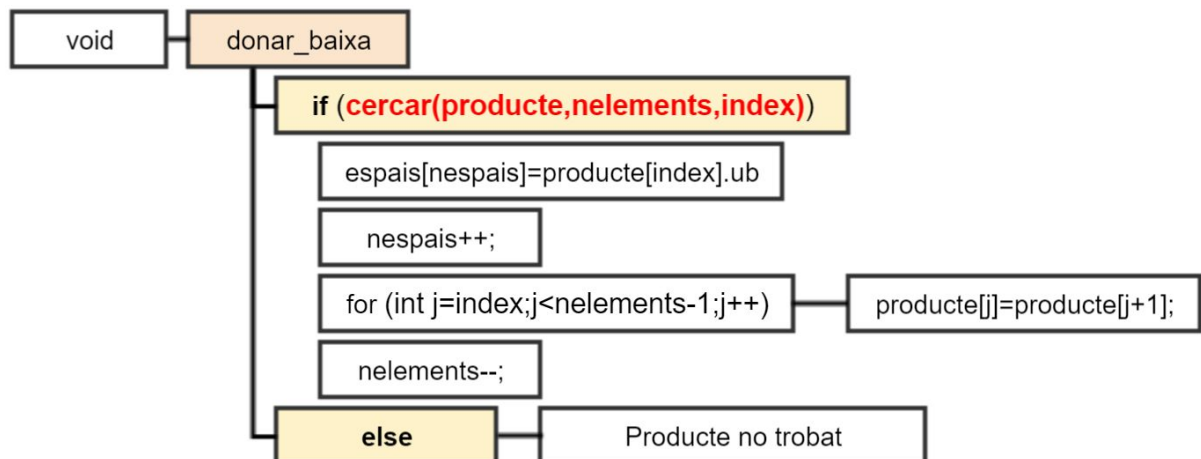
assigna la posició per defecte del producte. Com s'ha esmentat anteriorment, considerem que les primeres plantes, passadissos i seccions estan més a prop de l'entrada i, consegüentment, són llocs preferibles a l'hora de col·locar un nou article. Per tant, primer es comprova si hi ha algun lloc buit abans de l'últim lliure (vector “espais\_buits”). En cas afirmatiu (*if*), es col·loca en un d'aquests i es treuen les coordenades del vector. En cas contrari (*else*), se li assigna el lloc consecutiu al producte més allunyat i guardem la posició que haurà de tenir el producte consecutiu.

Cal fer notar que si “ultim\_lliuere” arribés a superar

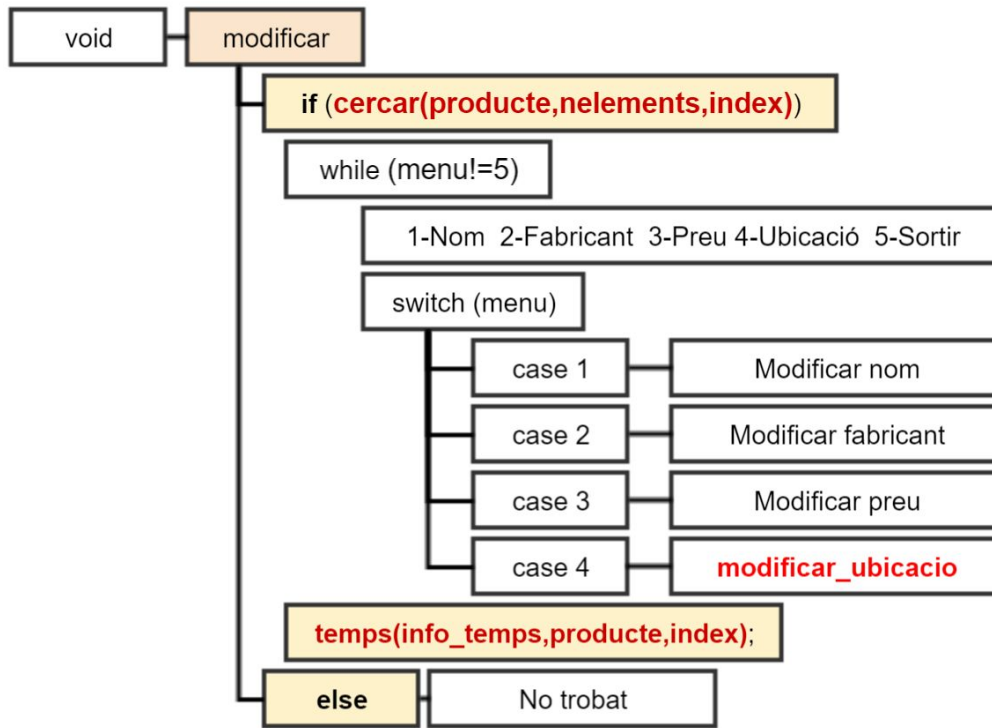
el màxim de plantes que hi ha al magatzem, o bé hi ha espais buits en algun lloc del magatzem o bé aquest estaria del tot ple i, per tant, no s'hauria pogut afegir cap article en primer lloc.



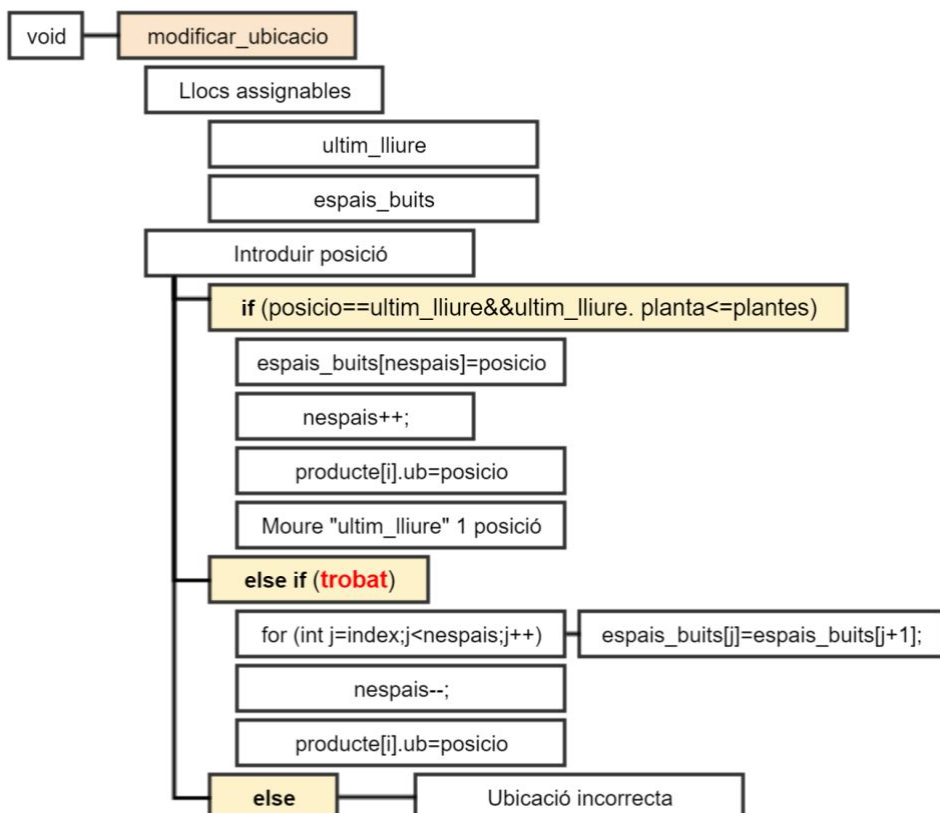
A continuació tenim “**donar\_baixa**”: aquesta acció demana el codi del producte a eliminar i, si el codi és d'algun dels articles, l'elimina del vector. Només cerca segons aquest criteri per si fos el cas que hi haguessin productes amb igual nom o fabricant. No s'han de confondre “cercar”, “cercar\_mostrar” i “cerca\_binaria”.



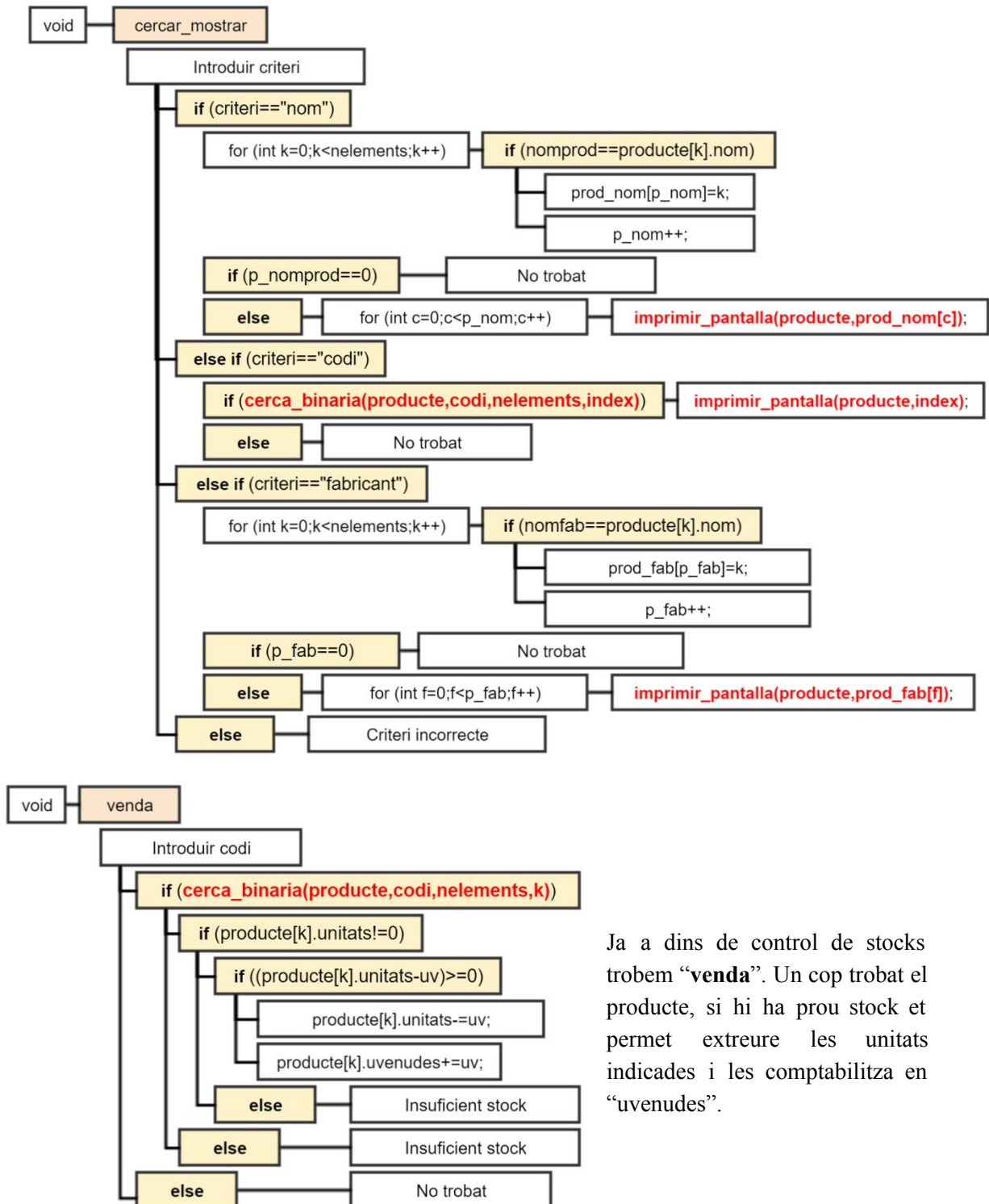
Pels mateixos motius que a “donar\_baixa”, l’acció “modificar” sols es pot dur a terme introduint el codi de l’article en qüestió. Es poden modificar el nom, el fabricant, el preu i la ubicació dins el magatzem. Un cop realitzats els canvis desitjats (opció 5), es guarda l’hora en què s’acaben les modificacions i es torna al menú de gestió de productes.



“**modificar\_ubicacio**” permet a l’usuari col·locar un ítem en alguna de les posicions disponibles més properes, és a dir, o bé “ultim\_lliure” o bé alguna posició de “espais\_buits”. “moure ultim\_lliure 1 posició” és un resum del què ja s’ha vist a “determinar\_posicio”.



“**cercar\_mostrar**” és una eina que mostra la fitxa completa dels articles d’igual nom, codi o fabricant a l’introduït. Com que el nom i el fabricant poden no ser únics, es crea un vector per tal de guardar tots els que coincideixin.



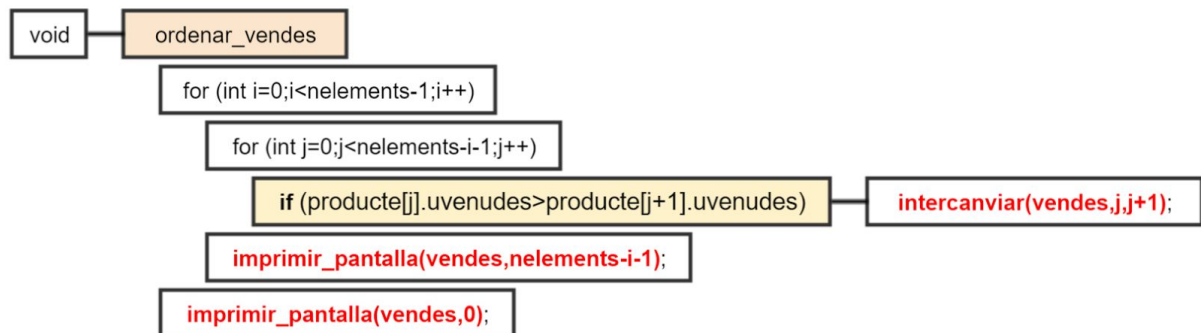
Ja a dins de control de stocks trobem “**venda**”. Un cop trobat el producte, si hi ha prou stock et permet extreure les unitats indicades i les comptabilitza en “uvenudes”.



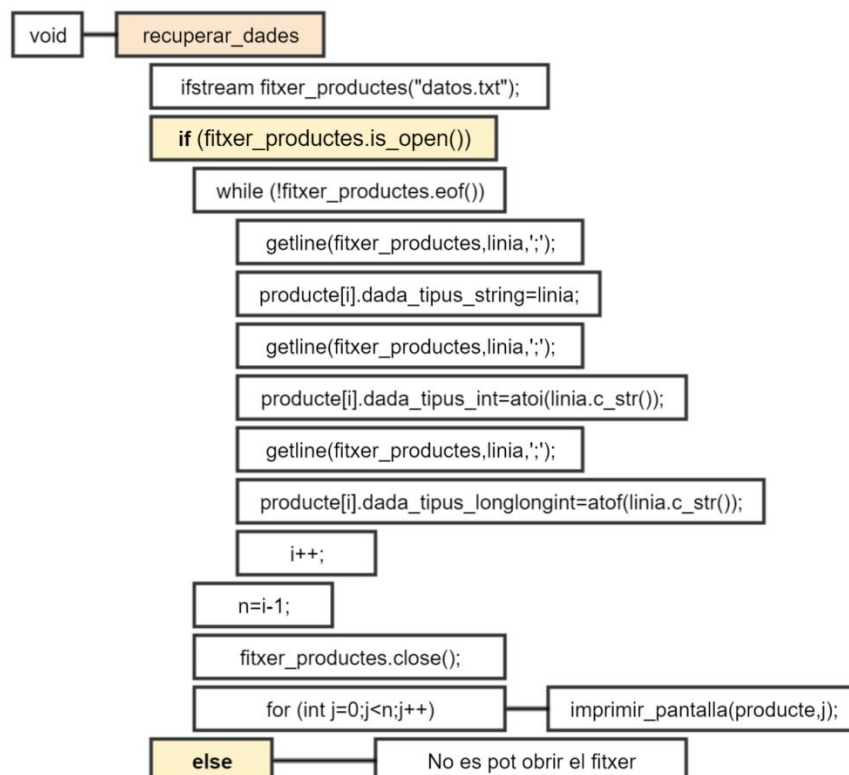
“**llista\_vendes**” mostra les fitxes de tots els articles ordenats en ordre no creixent segons el seu nombre de vendes. Dins l’acció es crea una còpia del vector “productes” i, a mesura que s’ordena amb “ordenar\_vendes”, es mostra per la pantalla.



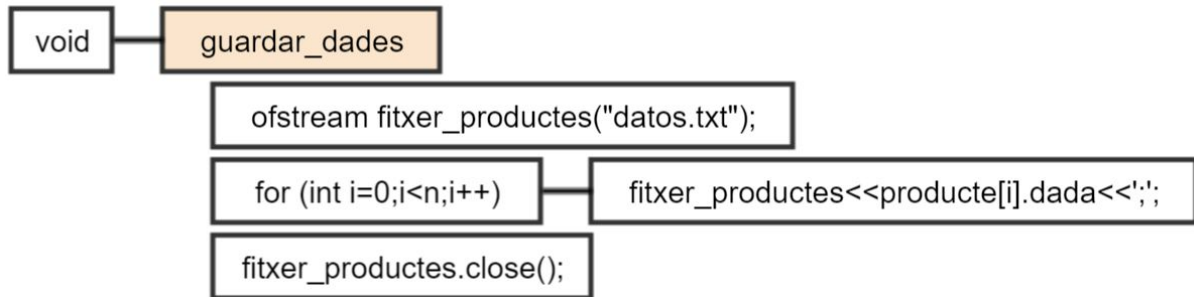
L’acció “**ordenar\_vendes**” ordena el vector amb el mètode bombolla de menor a major nombre de vendes. Com que després de cada passada el darrer element segur que és el major, per fer-ho més eficient s’aprofita per mostrar-lo per pantalla.



“**recuperar\_dades**” carrega la informació del fitxer “datos.txt” al programa sobre els productes guardats d’una sessió anterior. Per tal de fer això obre el fitxer i va copiant les dades. També les mostra per la pantalla, perquè l’usuari pugui tenir de seguida una idea global de què hi ha. L’arxiu amb què ve el programa no està en blanc, sinó que guarda la informació d’un parell de productes com a exemple que l’usuari pot mirar com a referència.



“**guardar\_dades**” sobreescriu en el fitxer “datos.txt” la informació sobre els productes que hi ha al programa en el moment en què s’acciona.



## 5.PROVES I POSTA A PUNT

Com és habitual en un projecte de la complexitat d’aquest, el projecte s’ha anat realitzant en diferents fases i ha estat necessari fer diferents proves de cada apartat abans que el projecte pogués unir-se i funcionar a la perfecció. Les divisions més clares van ser la dels blocs I, II, III i IV. En el bloc II, no van ser necessàries subdivisions, ja que és relativament curt, tot i en que la cerca binària es va realitzar bastants programes de prova. En el cas de el bloc I, es van fer proves de les accions de donar d’alta, donar de baixa i modificar, així com de la generació de codis EAN 13. Específicament l’acció de donar d’alta va portar, només per ella mateixa, bastants altres proves, com l’ordenació o la distribució dels productes per la seva posició. En el bloc III, també es van realitzar accions de prova per a cada apartat: llistar productes, vendes, comandes i llistar per unitats venudes. Per últim, en el bloc IV, només es va diferenciar per recuperar i desar dades en un fitxer.

Ja que havíem provat cada acció per separat i n’havíem comprovat el funcionament, a l’hora d’aglutinar totes les accions i d’estructurar el programa principal no hi va haver gaires dificultats i ho vam tenir llest amb força rapidesa.

## 6.CONCLUSIONS

Com a conclusions del treball i mirant els objectius plantejats a l’inici d’aquest, podem afirmar que hem realitzat el projecte amb èxit, ja que creiem que hem complert totes les metes proposades i hem resolt l’enunciat del treball. Fent un repàs als objectius inicials ens adonem que hem resolt tots els problemes del programa i hem après els nous conceptes de codis EAN 13, fitxers i llibreria *time.h*. Hem intentat fer un programa el més eficient possible, però som conscients que el tema eficiència seria un apartat amb possibilitats de millora, així com implementar altres funcionalitats al programa que permetessin fer-lo més interactiu i atractiu per a l’usuari. Tot i això, pensem que el programa està ben resolt ja que compleix tots els requisits inicials demanats a l’enunciat.

## 7.AUTOAVALUACIÓ

Pel que fa la manera de repartir-se la feina, pensem que s'ha fet d'una forma equitativa i organitzada. La Marina Navarro s'ha encarregat de la generació de codis EAN 13, ha ideat la manera de determinar la ubicació dels productes i ha introduït la llibreria *time.h* al programa. Pel que fa la Núria Martín s'ha encarregat dels bloc III, així com tots els apartats relacionats amb els fitxers: carregar i guardar dades i llegir els codis dels països per a la generació de codis EAN 13. Conjuntament, però, s'ha realitzat el bloc II i la resta d'apartats del bloc I. En el cas de la memòria, també s'ha fet conjuntament.

Al llarg del treball ens hem anat trobant amb dificultats i dubtes que no ens permetien continuar amb la realització del projecte. Una de les majors dificultats va ser la implementació dels fitxers, ja que era un concepte nou i el necessitàvem al principi del nostre programa (generació del codi EAN 13), l'ordenació dels codis també va suposar una dificultat, ja que es vam provar diferents mètodes: inserció, selecció i bombolla. Finalment es va optar pel mètode bombolla, ja que era el que funcionava millor. Pel que fa als dubtes que han anat sorgint al llarg del treball, podríem destacar els dos més importants:

- **La ubicació dels productes.** Des d'un bon principi vam tenir dubtes sobre aquest apartat i conseqüentment vam provar diferents maneres d'ubicar els productes, sobretot amb taules. A més cal remarcar que ho volíem fer d'una manera organitzada i lògica. Finalment, vam optar per una tupla d'ubicació que anés dins la tupla de productes i un vector d'espais buits. D'aquesta manera, primer omplíem els espais buits del magatzem (de productes que s'haguessin donat de baixa) amb els nous productes (per tal de no tenir espais buits al llarg del magatzem) si aquest no era el cas, buscàvem l'última posició lliure i hi col·locàvem l'article.
- **Criteris de cerca en les accions de donar de baixa i modificar.** Pensant que al bloc II, ja s'usava un criteri de cerca, vam pensar en reutilitzar l'acció a l'hora de seleccionar el producte a donar de baixa o modificar. Més tard però ens vam adonar que un fabricant podia tenir més d'un producte al catàleg i que hi podia haver diferents productes amb el mateix nom i que, per tant, aquests serien uns criteris poc precisos. D'aquesta manera, vam decidir que el millor criteri de cerca en aquestes accions seria el codi, ja que és un número únic de cada producte.

## 8.BIBLIOGRAFIA

Els documents consultats al llarg del projecte són els següents:

- XHAFA XHAFA, Fatos. *Programació en fitxers C++*. [pdf]. Disponible a [www.atena.upc.edu/moodle](http://www.atena.upc.edu/moodle)
- XHAFA XHAFA, Fatos. *Llibreria time.h de C++*. [pdf]. Disponible a [www.atena.upc.edu/moodle](http://www.atena.upc.edu/moodle)
- XHAFA XHAFA, Fatos. *Generació de codis EAN 13*. [pdf]. Disponible a [www.atena.upc.edu/moodle](http://www.atena.upc.edu/moodle)