

Semestre 2 – Pràctica 1 The Market



Índex

1. INTRODUCCIÓ	3
2. OBJECTIUS.....	4
3. FUNCIONAMENT.....	5
3.1 Opció 1: Càrrega de productes d'un proveïdor.....	6
3.2 Opció 2: Llistar productes ordenats per estoc	9
3.3 Opció 3: Buscar producte per nom	11
3.4 Opció 4: Sortir	12
4. CONSIDERACIONS.....	13
5. REQUISITS MÍNIMS	14
6. DATA DE LLIURAMENT I AVALUACIÓ.....	15

1. INTRODUCCIÓ

Un supermercat ens ha demanat que desenvolupem una aplicació que permeti introduir els productes que arriben de diferents proveïdors. També ens ha demanat que desenvolupem altres funcionalitats relacionades, com llistar tots els productes que s'hagin carregat fins el moment.

2. OBJECTIUS

Els objectius bàsics d'aquesta pràctica son:

- Practicar l'obertura i tancament de fitxers.
- Realitzar operacions bàsiques de lectura/escriptura sobre fitxers.
- Comprendre les diferències entre un fitxer de text i un binari.
- Seguir consolidant les bones pràctiques d'estructuració de codi en procediments i funcions.

3. FUNCIONAMENT

Quan iniciem el programa, el primer que es farà es donar la benvinguda i mostrar un menú on es podrà escollir quin tipus de tasca es vol realitzar. Aquest menú serà el que considerarem el “Menú principal” (vegeu Output 1).

```
Welcome!  
  
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit  
Select option: █
```

Output 1: Missatge de benvinguda i menú principal..

Les diferents tasques que es poden realitzar són: 1) Carregar informació de proveïdors. 2) Llistar productes per ordre. 3) Buscar un producte per nom i mostrar informació relacionada. 4) Sortir del programa.

Després d'executar una opció, s'haurà de tornar a mostrar el menú i demanar de nou la tasca que es vol realitzar fins que no es seleccioni l'opció 4.

Mai s'haurà de finalitzar l'execució del programa si no s'ha seleccionat l'opció corresponent (opció 4).

Si l'usuari introdueix un valor incorrecte (en qualsevol menú), s'haurà de mostrar un missatge informatiu i tornar a demanar que s'introdueixi l'opció desitjada (vegeu Output 2).

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit  
Select option: 5  
  
ERROR: Wrong option number  
  
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit  
Select option: █
```

Output 2. Exemple d'execució d'una opció incorrecta.

3.1 Opció 1: Càrrega de productes d'un proveïdor

El supermercat ens ha comentat que necessita carregar informació de tres proveïdors principals i aquesta informació consisteix en el nom del producte, l'estoc i el preu. Quan l'usuari en el menú principal seleccioni la opció 1, es mostrarà un altre menú que permetrà carregar la informació d'un dels tres proveïdors (vegeu output 3).

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: 1

1. Supplier 1 | 2. Supplier 2 | 3. Supplier 3
Choose one to load from: █
```

Output 3. Menú per carregar la informació del proveïdor.

La informació d'aquests productes vindrà en diferents fitxers que s'hauran d'emmagatzemar correctament a memòria.

S'ha de tenir en compte que el nom del producte pot contenir espais, l'estoc del producte sempre serà un enter i el preu és un número decimal.

En cas que ens els diferents fitxers existeixin productes repetits (amb el mateix nom), s'afegirà cada estoc a l'estoc existent i el nou preu del producte serà el més alt entre el que ja teníem emmagatzemat (en el nostre programa) i el que volem afegir. Així doncs, si un producte anomenat "t-shirt" apareix en dos dels fitxers, un amb un estoc == 3 i preu == 10.4, i l'altre amb estoc == 5 i preu == 5, l'estat final del producte "t-shirt" en el supermercat serà d'un estoc == 8 i preu == 10.4.

En cas que algun producte aparegui amb estoc == 0 o que el seu nom contingui una '@', aquest producte no es carregarà (vegeu apartat Opció 2: Llistar productes ordenats per estoc).

Cadascun dels proveïdors envia la informació dels productes amb formats de fitxers diferents.

Per a cadascun dels proveïdors, es demanarà el nom del fitxer (sense l'extensió). El nostre programa haurà d'afegir l'extensió ".txt" al nom introduït per l'usuari per als dos primers proveïdors i l'extensió ".bin" per al tercer.

Així doncs, el proveïdor 1 (vegeu Output 4), envia la informació en un fitxer de text (al seu nom li afegirem l'extensió ".txt") on hi hauran múltiples dades amb el següent format:

```
<stock> <price> <product name>
```

Exemple:

```
25 3.45 red t-shirt
```

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: 1

1. Supplier 1 | 2. Supplier 2 | 3. Supplier 3
Choose one to load from: 1

Type the name of the file (without '.txt'): supplier1

1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: █
```

Output 4. Càrrega de productes del proveïdor 1.

El proveïdor 2, envia la informació en un fitxer de text (al seu nom també afegirem l'extensió ".txt"). Aquest fitxer estarà format amb dades amb el següent format:

```
Product name: <product name>
Price: <price> euros
Stock: <stock>
```

Exemple:

```
Product name: red t-shirt
Price: 3.45 euros
Stock: 25
```

El proveïdor 3, envia la informació en un fitxer binari (al seu nom li afegirem l'extensió ".bin") (vegeu Output 5).

```
1. Supplier 1 | 2. Supplier 2 | 3. Supplier 3
Choose one to load from: 3

Type the name of the file (without '.bin'): supplier3

1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: █
```

Output 5. Càrrega de productes del proveïdor 3.

El fitxer binari conté múltiples registres definits amb el següent tipus:

```
#define MAX_STRING_LENGTH 70

typedef struct {
    char name[MAX_STRING_LENGTH];
    int stock;
    float price;
} Product;
```

Si després de completar el nom del fitxer amb l'extensió de qualsevol proveïdor, aquest no existeix, es mostrarà un missatge per pantalla i es tornarà al menú principal (vegeu Output 6).

```
1. Supplier 1 | 2. Supplier 2 | 3. Supplier 3
Choose one to load from: 3

Type the name of the file (without '.bin'): somefile

ERROR: Can't open file 'somefile.bin'

1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: █
```

Output 6. Si el nom del fitxer no existeix, es mostra un missatge d'error i es torna al menú principal.

Una vegada s'hagi carregat el fitxer desitjat, es tornarà al menú principal (vegeu Output 4). Es podran carregar fitxers tantes vegades com desitgi l'usuari, però mai tindrem més de 30 productes diferents.

3.2 Opció 2: Llistar productes ordenats per estoc

La segona tasca que podrà realitzar l'usuari del nostre programa és llistar productes ordenats per estoc de forma ascendent. En aquest cas, el programa mostrarà per pantalla la informació de cada producte (vegeu Output 7).

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: 2

Product #1:
Name: light bulb
Stock: 1 units
Price: 10.00 euros
---
Product #2:
Name: soap
Stock: 3 units
Price: 20.60 euros
---
Product #3:
Name: sneakers
Stock: 6 units
Price: 60.90 euros
---
Product #4:
Name: jeans
Stock: 12 units
Price: 15.00 euros
---
Product #5:
Name: t-shirt
Stock: 13 units
Price: 10.40 euros
---
Product #6:
Name: ball
Stock: 15 units
Price: 5.00 euros
---
Product #7:
Name: jacket
Stock: 20 units
Price: 20.00 euros
---

1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: █
```

Output 7. Llistat dels productes després de carregar la informació.

En el cas que el supermercat no disposi de cap producte (o encara no s'hagi llegit cap fitxer de proveïdor), es mostrarà un missatge per pantalla (vegeu Output 8).

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit  
Select option: 2
```

```
There are 0 products in the store
```

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit  
Select option: █
```

Output 8. Llistat de productes. No s'ha carregat cap fitxer o el que s'ha intentat carregar només contenia productes amb estoc == 0 o el seu nom contenia una '@'.

3.3 Opció 3: Buscar producte per nom

Aquesta opció permet buscar un producte pel seu nom. Si es troba el producte, es mostrarà la seva informació per pantalla i es tornarà al menú principal (vegeu Output 9). En cas que no es trobi, es mostrarà un missatge i es tornarà al menú principal (vegeu Output 10).

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: 3

Product name?: ball

Product found!
Name: ball
Stock: 15 units
Price: 5.00 euros

1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: █
```

Output 9. Cerca d'un producte. Es troba el producte en qüestió.

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: 3

Product name?: te@

ERROR: Product not found

1. Load products from supplier | 2. List products | 3. Show product | 4. Exit
Select option: █
```

Output 10. Cerca del producte. No es troba el producte en qüestió.

3.4 Opció 4: Sortir

L'última opció simplement mostrarà un missatge de comiat i es finalitzarà l'execució del programa (vegeu Output 11).

```
1. Load products from supplier | 2. List products | 3. Show product | 4. Exit  
Select option: 4
```

```
Bye!
```

Output 11. Missatge de comiat.

4. CONSIDERACIONS

Per la implementació d'aquesta pràctica s'han de tenir en compte les següents consideracions:

1. S'ha de seguir el format mostrat en els exemples.
2. La forma de gestionar els errors és estrictament la mostrada a l'enunciat.
3. El programa ha d'estructurar-se correctament en procediments i funcions. Un programa amb el codi estructurat de manera errònia no serà acceptat.
4. En les sortides per pantalla es deixarà com a màxim una línia en blanc. És exactament el format mostrat en els exemples.
5. Podeu assumir que els fitxers no estaran buits i seguiran el format especificat, tot i que s'han de controlar els errors indicats.
6. Disposareu dels fitxers d'exemple usats en les imatges d'aquest enunciat per a comprovar el funcionament del programa. Aquests fitxers estan generats utilitzant la plataforma matagalls/CodeRunner.

5. REQUISITS MÍNIMS

Aquesta pràctica ha de complir amb una sèrie de requeriments de qualitat mínims per poder ser avaluada:

- El codi ha d'estar correctament comentat de manera que sigui llegible sense dificultat.
- S'ha de seguir la guia d'estils de l'assignatura.
- Per poder aprovar és necessari que la pràctica superi satisfactòriament els tests de CodeRunner.
- El programa haurà de desenvolupar-se íntegrament en l'entorn Linux de Matagalls, utilitzant l'editor vim per escriure el codi en C i el compilador gcc per a la obtenció de l'executable corresponent.
- No es pot utilitzar cap eina ni instrucció de C que no s'hagi explicat a classe.
- El codi ha d'estar estructurat correctament utilitzant un mínim de 6 procediments o funcions significatius. En aquest sentit, es valorarà especialment que en el codi no es repeteixin conjunts de sentències.
- No s'acceptarà cap pràctica que no compleixi la normativa de pràctiques.

6. DATA DE LLIURAMENT I AVALUACIÓ

La data de lliurament d'aquesta pràctica per poder obtenir la màxima qualificació es el 13 de Març de 2022.

Per tal que la pràctica es consideri **lliurada** s'hauran de complir les següents condicions:

1. Executar el codi en el CodeRunner habilitat expressament i superar satisfactòriament tots els tests.
2. Entregar en el pou corresponent un fitxer .c amb el codi de la pràctica.

Es recorda que la pràctica serà avaluada de la següent manera:

- Execució: Té un pes del 80% i avalua el correcte funcionament de la pràctica. Serà la qualificació obtinguda en els tests del CodeRunner.
- Qualitat SW: Té un pes del 20% i s'avaluarà la qualitat del codi entregat i el seguiment de la Guia d'Estils de Programació que teniu disponible a l'eStudy.

Important: El fet de superar únicament els tests no implica que la pràctica estigui aprovada.