

## Semestre 2 - Pràctica 3 myDoctor expert



## Índex

1. Introducció.....	3
2. Funcionament.....	4
2.1 Opció 1: Show diseases .....	7
2.2 Opció 2: Load research advances.....	8
2.3 Opció 3: Help me!.....	10
2.4 Opció 4: Quit .....	13
3. Procediment de compilació i lliurament .....	14
4. Consideracions.....	15
5. Tests per la memòria dinàmica .....	16
6. Dates de lliurament i avaluació .....	17

## 1. Introducció

A la darrera pràctica s'ha vist que el nombre de símptomes d'una malaltia pot variar al llarg de l'execució de myDoctor, fet que ha provocat una reflexió sobre el disseny aplicat a l'hora d'emmagatzemar aquesta informació. Així, s'ha confirmat que l'aplicació ha de patir alguns canvis per tenir en compte correctament aquesta circumstància. Amb aquest nou lliurament de myDoctor (versió Expert), s'optimitzarà la manera de desar la informació relacionada amb els símptomes de cada malaltia, fent-la més flexible i versàtil.

### OBJECTIU

L'objectiu final d'aquesta pràctica és posar en pràctica els coneixements adquirits referents a les diferents estructures de dades lineals vistes a classe, així com al concepte de TAD. Alhora, s'aprofita la pràctica per revisar altres temàtiques prèviament estudiades. Concretament, es pretén :

- Consolidar el concepte de TAD
- Integrar i usar correctament una estructura de dades lineal
- Consolidar l'ús de memòria dinàmica
- Consolidar l'ús de sentències i algorísmica bàsica
- Consolidar els coneixements sobre l'estructuració d'un programa
- Consolidar l'ús de procediments, funcions i pas de paràmetres
- Consolidar l'ús de fitxers de text i binaris
- Revisar diferents algorismes d'ordenació estàndards
- Consolidar la compilació manual d'un projecte en C

## 2. Funcionament

Com a la pràctica anterior, el programa començarà carregant en el sistema el conjunt de malalties conegudes, sol·licitant el nom del fitxer que les conté (veure Outputs 1 i 2).

Aquest fitxer serà un fitxer de text que seguirà el format del següent exemple:

```
2      // número de malalties en el fitxer
Flu
3
Weakness (8)
Fever (9)
General pain (5)
Bronchitis
2
Cough (9)
Fever (6)
```

Com es pot observar, en aquest fitxer apareixeran cadascuna de les malalties a carregar en el nostre programa. Com a la pràctica anterior, la primera línia del fitxer estarà formada per un enter, que indicarà quantes malalties existeixen al fitxer. Per a cada malaltia, apareix el nom de la malaltia, un enter que indica el nombre de símptomes associats i després, per a cada símptoma, apareix el nom del símptoma i, entre parèntesi, la rellevància d'aquest símptoma en la malaltia (serà un valor sencer entre 1 i 10). Totes les malalties i tots els símptomes poden estar formats per més d'una paraula, encara que mai se superaran els 50 caràcters. No es pot suposar cap nombre màxim de malalties ni de símptomes per malaltia.

En cas que el fitxer no existeixi s'haurà de mostrar un missatge d'error i el programa haurà de finalitzar (Output 1). Si el fitxer existeix, es carregarà tota la informació a memòria, es donarà el missatge de benvinguda i es mostrarà el menú principal de l'aplicació (Output 2).

```
Enter diseases file name: file45
ERROR: Diseases file not found
```

*Output 1. S'introdueix el nom d'un fitxer que no existeix. El programa mostra el missatge i finalitza.*

```
Enter diseases file name: diseases
Welcome to myDoctor!

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: █
```

*Output 2. S'introdueix un fitxer correctament, es dona la benvinguda i es mostra el menú principal.*

L'usuari del programa podrà introduir qualsevol valor (sigui numèric o no) i en cas de no coincidir amb cap de les opcions del programa, es mostrarà un missatge d'error i es tornarà a mostrar el menú, sol·licitant que torni a introduir un valor (Output 3).

```
Enter diseases file name: diseases
Welcome to myDoctor!

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: *
ERROR: Wrong option

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: hello
ERROR: Wrong option

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: █
```

*Output 3. Es produeixen diferents errors i s'informa l'usuari que l'opció triada és incorrecta.*

Després d'executar qualsevol opció del menú (excepte l'opció 4), el programa ha de tornar al menú principal. Com veurem més endavant, si s'executa l'opció 4, el programa ha de finalitzar.

Per poder guardar de manera òptima les dades de l'aplicació, cal crear diferents estructures dinàmiques per poder emmagatzemar-les:

Com que el nombre de malalties no varia al llarg de l'execució del programa, la creació de l'estructura de dades es continuarà realitzant a l'inici del programa, amb memòria dinàmica, utilitzant de referència el valor que apareix a la primera línia del fitxer.

En canvi, per emmagatzemar els símptomes, ho farem mitjançant una estructura de dades lineal. Concretament, cada malaltia disposarà d'una LinkedList on hi guardarem els símptomes. A mode orientatiu, i com a ajuda, el tipus Disease podria ser definit d'aquesta manera:

```
typedef struct {
    char disease[MAX_CHAR];
    int n_symptoms;
    LinkedList symptoms;
} Disease;
```

Independentment de com es dissenyi aquest tipus, la LinkedList utilitzada estarà formada obligatòriament per elements de tipus Symptom, definits així:

```
#define MAX_CHAR      50
typedef struct {
    char symptom[MAX_CHAR];
    int relevance;
} Symptom;

typedef Symptom Element;
```

Aquest tipus s'ha d'incloure al fitxer linkedlist.h i es garanteix que també estarà inclòs a la LinkedList usada a CodeRunner.

Es pot fer servir de base la LinkedList que es troba disponible a l'apartat de Continguts de l'eStudy.

## 2.1 Opción 1: Show diseases

L'opció "Show diseases" no canvia respecte a la pràctica anterior i permet mostrar per pantalla tota la informació registrada actualment a memòria (malalties, símptomes i rellevància d'aquests). Com es pot observar en l'Output 4, cada malaltia es mostrarà en una línia, indicant entre parèntesi els símptomes associats i la seva rellevància. Cada símptoma i la seva rellevància se separa pel símbol '#' i els diferents símptomes se separen pel símbol '|' amb un espai a cada costat.

```
Enter diseases file name: diseases
Welcome to myDoctor!

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: 1

Registered diseases:

Flu (Weakness#8 | Fever#9 | General pain#5)
Bronchitis (Cough#9 | Fever#6)
Pneumonia (Cough#10 | Nausea#2 | Fatigue#4 | Muscle pain#5)
Fungal nail infection (Infected skin#6 | Nail discolouration#8)

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: █
```

*Output 4. L'opció 1 del programa permet mostrar tota la informació actual relacionada amb les malalties, símptomes i la seva rellevància.*

Després de mostrar la informació, es torna a mostrar el menú principal, esperant que l'usuari introdueixi la nova opció.

Els valors que es mostren en els exemples són els que s'obtenen utilitzant els fitxers de prova adjuntats a l'enunciat de la pràctica.

## 2.2 Opció 2: Load research advances

L'opció "Load research advances" no canvia respecte a l'anterior pràctica i s'executa quan disposem de noves dades procedents de la recerca, que permeten afegir algun símptoma nou a alguna malaltia o bé actualitzar la rellevància d'un símptoma en una de les malalties existents.

En executar aquesta opció es demana el nom del fitxer que conté aquesta informació. Si el nom del fitxer no és correcte (el fitxer no existeix), s'haurà de mostrar un missatge d'error i es tornarà al menú principal (Output 5).

```
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 2
Enter research file name: hello
ERROR: Research file not found
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: █
```

*Output 5. El fitxer amb les dades procedents de la recerca, no existeix.*

Si el fitxer existeix, el programa carregarà aquesta informació. Aquest fitxer és un fitxer binari format per registres del següent tipus:

```
#define MAX_CHAR      50

typedef struct {
    char disease[MAX_CHAR];
    char symptom[MAX_CHAR];
    int relevance;
} FileDisease;
```

En cas que el fitxer contingui alguna malaltia que no està prèviament registrada en el programa, s'ignorarà. En cas que aparegui un nou símptoma, s'afegirà com a símptoma nou per a la malaltia indicada, amb el seu valor de rellevància indicat. Cal tenir en compte que aquest cas pot suposar una afectació en l'estructura dinàmica. Finalment, en cas que la malaltia i el símptoma existeixin, s'actualitzarà el valor de rellevància amb el nou valor indicat en el fitxer (en cas que sigui el mateix valor, s'actualitzarà igualment).

Una vegada carregada aquesta informació, es mostrarà per pantalla (veure Output 6) quantes malalties s'han ignorat (perquè no existien en el programa), quants símptomes s'han actualitzat i quants símptomes s'han afegit a alguna malaltia (pot haver-hi més d'una actualització per a una mateixa malaltia).



```
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 2
Enter research file name: research

1 diseases ignored
1 symptoms updated
3 symptoms added

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: █
```

*Output 6. Una vegada carregat el fitxer amb les dades de recerca, s'indiquen quantes malalties s'han ignorat i quants símptomes s'han actualitzat i afegit.*

A continuació teniu un exemple complet (Output 7) on es carrega la informació del fitxer de prova “diseases”, es mostra la informació de les malalties (opció 1), es carrega el fitxer de prova “research” (opció 2) i es torna a mostrar la informació actualitzada (opció 1). Els dos fitxers estan disponibles juntament amb l'enunciat d'aquesta pràctica.

```
Enter diseases file name: diseases
Welcome to myDoctor!

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 1

Registered diseases:

Flu (Weakness#8 | Fever#9 | General pain#5)
Bronchitis (Cough#9 | Fever#6)
Pneumonia (Cough#10 | Nausea#2 | Fatigue#4 | Muscle pain#5)
Fungal nail infection (Infected skin#6 | Nail discolouration#8)

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 2
Enter research file name: research

1 diseases ignored
1 symptoms updated
3 symptoms added

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 1

Registered diseases:

Flu (Weakness#8 | Fever#7 | General pain#5 | Cough#5)
Bronchitis (Cough#9 | Fever#6)
Pneumonia (Cough#10 | Nausea#2 | Fatigue#4 | Muscle pain#5 | Rapid heartbeat#2)
Fungal nail infection (Infected skin#6 | Nail discolouration#8 | Fever#1)

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: █
```

*Output 7. Exemple complet on es carrega el fitxer inicial, es mostra la informació, es carrega el fitxer de recerca i es torna a mostrar la informació actualitzada.*

### 2.3 Opció 3: Help me!

L'opció 3 del programa permet informar l'usuari de quines possibles malalties pot tenir a partir dels símptomes que pateix. El primer que farà el programa és demanar a l'usuari que introdueixi els seus símptomes mitjançant una cadena de text, separant cada símptoma pel caràcter '+'. Es pot suposar que el format de la cadena sempre serà correcte i que com a molt s'entraran 10 símptomes.

Una vegada introduïts els símptomes es mostraran per pantalla les possibles malalties que pot patir l'usuari. Per a descobrir-ho, s'analitzaran els símptomes descrits per l'usuari i els símptomes existents per a cada malaltia. Si l'usuari pateix un dels símptomes d'una malaltia, se sumarà la rellevància d'aquest símptoma a aquesta malaltia. Al final, l'objectiu és disposar d'un llistat amb les malalties que tinguin una puntuació superior a 0 (que coincideixi algun símptoma) obtingut a partir de la suma de totes la rellevàncies dels símptomes coincidents amb els de l'usuari.

Aquest llistat es mostrarà ordenant les malalties de major a menor segons la seva puntuació associada (per a mostrar primer les que tenen més probabilitats que l'usuari pateixi). Per a cada malaltia es mostra el seu total de rellevància. Es pot veure un exemple en l'Output 8. Cal tenir en compte que no hi ha un màxim de malalties possibles en el sistema, per la qual cosa aquest llistat haurà de generar-se utilitzant una estructura de dades dinàmica.

```
Enter diseases file name: diseases
Welcome to myDoctor!

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: 1

Registered diseases:

Flu (Weakness#8 | Fever#9 | General pain#5)
Bronchitis (Cough#9 | Fever#6)
Pneumonia (Cough#10 | Nausea#2 | Fatigue#4 | Muscle pain#5)
Fungal nail infection (Infected skin#6 | Nail discolouration#8)

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: 3
Enter your symptoms: Fever+Cough

Most probable diseases:
Total relevance 15: Bronchitis
Total relevance 10: Pneumonia
Total relevance 9: Flu

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: █
```

*Output 8. Primer es carrega el fitxer de malalties i després es mostra el seu contingut. Finalment, mitjançant l'opció 3, es mostra un llistat ordenat de les malalties més probables que pot patir l'usuari.*

En cas que s'introdueixi un símptoma que no existeix en el nostre programa, aquest símptoma no tindrà cap afectació.

Per a realitzar l'ordenació de les dades, es pot usar qualsevol mètode d'ordenació per a tipus de dades estructurades. Es considerarà que mai hi haurà empat en la suma total de rellevància entre dues o més malalties. Adjunt a aquest enunciat es disposa d'un document on es pot revisar el funcionament d'alguns d'aquests mètodes.

Si no hi hagués cap malaltia possible amb els símptomes descrits, llavors s'ha de mostrar un missatge indicant aquesta circumstància (veure Output 9).

```
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 1
Registered diseases:
Flu (Weakness#8 | Fever#9 | General pain#5)
Bronchitis (Cough#9 | Fever#6)
Pneumonia (Cough#10 | Nausea#2 | Fatigue#4 | Muscle pain#5)
Fungal nail infection (Infected skin#6 | Nail discolouration#8)
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 3
Enter your symptoms: Wheezing+Rapid heartbeat
No diseases found!
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: █
```

*Output 9. Els símptomes no coincideixen amb cap malaltia registrada.*

A l'Output 10 es mostra un altre exemple complet carregant els dos fitxers de prova de la pràctica (text i binari) i introduint un símptoma que no existeix. En qualsevol cas, es recomana crear altres fitxers de prova per a provar el correcte funcionament de la pràctica.

```
Enter diseases file name: diseases
Welcome to myDoctor!

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: 2
Enter research file name: research

1 diseases ignored
1 symptoms updated
3 symptoms added

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: 1

Registered diseases:

Flu (Weakness#8 | Fever#7 | General pain#5 | Cough#5)
Bronchitis (Cough#9 | Fever#6)
Pneumonia (Cough#10 | Nausea#2 | Fatigue#4 | Muscle pain#5 | Rapid heartbeat#2)
Fungal nail infection (Infected skin#6 | Nail discolouration#8 | Fever#1)

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: 3
Enter your symptoms: Nail discolouration+Wheezing+General pain+Fever

Most probable diseases:
Total relevance 12: Flu
Total relevance 9: Fungal nail infection
Total relevance 6: Bronchitis

1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit

Enter option: █
```

*Output 10. Exemple on es carreguen tots dos fitxers de prova i s'executa l'opció 3. A continuació s'introdueixen un conjunt de símptomes (entre els quals un que no existeix) i es mostren les malalties possibles.*

## 2.4 Opció 4: Quit

L'opció 4 del programa mostra un missatge de comiat a l'usuari i finalitza l'execució del programa (veure Output 11).

```
1. Show diseases | 2. Load research advances | 3. Help me! | 4. Quit
Enter option: 4
Stay healthy!
```

*Output 11. La opció 4 finalitza l'execució del programa.*

### 3. Procediment de compilació i lliurament

Aquesta pràctica haurà de ser compilada mitjançant un fitxer makefile que permeti generar un executable a partir dels següents fitxers:

- fitxer .c principal (on es troba el main)
- linkedlist.c
- linkedlist.h

En el moment del lliurament cal enganxar el contingut del fitxer principal (main) a CodeRunner. El sistema ja portarà incorporada la LinkedList amb el tipus Symptom correctament definit, per la qual cosa no cal que s'enganxin els fitxers corresponents a l'estructura de dades lineal. Cal tenir en compte que com que el tipus Symptom ja estarà definit al fitxer linkedlist.h, aquest tipus no ha d'aparèixer al fitxer principal.

El fitxer makefile s'haurà de lliurar de la manera que s'explica a l'apartat 6 d'aquesta pràctica.

## 4. Consideracions

Per a la implementació de la pràctica cal tenir en compte les següents consideracions:

1. S'ha de seguir el format mostrat en els exemples.
2. En les sortides per pantalla es deixarà com a màxim una línia en blanc (observeu els exemples al llarg de l'enunciat).
3. La manera de gestionar els errors és estrictament la que demana l'enunciat. Si es diu que ha de tornar a introduir-se la dada no s'acceptarà que per exemple torni al menú inicial.
4. El codi ha d'estar correctament comentat de manera que sigui llegible sense dificultat. Comproveu que segueix la guia d'estils de l'assignatura.
5. El codi ha d'estar correctament estructurat en procediments i/o funcions. Es demana un mínim de 6 funcions o procediments (sense comptar el main) durant el desenvolupament de la pràctica. Es valorarà que l'objectiu, disseny i ús d'aquestes funcions/procediments siguin correctes (significança de les funcions o procediments, reaprofitament de codi, pas de paràmetres, etc).
6. Per tal que la pràctica sigui acceptada és necessari que superi satisfactòriament tots els tests de CodeRunner i tenir una nota de Qualitat del Software superior a 1. **No es considerarà aquesta pràctica com acceptada si se superen els tests però no s'utilitza la memòria dinàmica requerida en les especificacions de l'enunciat.**
7. El programa s'hauria de desenvolupar íntegrament a l'entorn Linux de Matagalls, utilitzant l'editor vim per escriure el codi C i el compilador gcc per obtenir l'executable corresponent. Allà la podeu provar abans de testejar-la al CodeRunner.
8. No es pot usar cap instrucció de C que no s'hagi explicat en classe.

## 5. Tests per la memòria dinàmica

Tal com s'indica en aquest enunciat, en aquesta pràctica s'haurà d'usar memòria dinàmica per a crear certes estructures de dades. És per això que, en realitzar els tests de CodeRunner, es comprovarà que això s'ha fet correctament. Concretament, si no es compleixen unes certes condicions, en el test apareixerà, a la columna "Got", un missatge d'error en la sortida del programa. Es distingeixen els següents casos:

- "ERROR: Some pointer exceeded its allocated memory": Aquest error apareixerà si s'ha accedit a alguna direcció de memòria que no ha estat reservada pel programa.
- "ERROR: Some pointers have not been deallocated": Aquest error apareixerà si no s'ha alliberat tota la memòria dinàmica en finalitzar el programa.
- També apareixerà un error si s'intenta alliberar memòria dinàmica ja alliberada (doble free).



## 6. Dates de lliurament i avaluació

La data de lliurament d'aquesta pràctica per poder obtenir la màxima qualificació és el 21 de maig de 2023.

Perquè la pràctica es consideri **lliurada** s'hauran de complir les condicions següents:

1. Executar el codi al CodeRunner habilitat expressament i superar satisfactòriament els tests. Consulteu l'apartat 3 per a més detall.
2. Lliurar al pou corresponent un fitxer .zip que contingui els diferents fitxers que conformen la pràctica (inclòs el makefile).
3. Lliurar una memòria final de les pràctiques d'aquest semestre, que inclogui:
  - a. Portada i índex
  - b. Breu resum de l'enunciat de cada pràctica
  - c. Estructuració (procs/funks) de les pràctiques 2.1, 2.2 i 2.3
  - d. Principals dificultats i solucions aplicades
  - e. Conclusions finals
  - f. Estimació del temps dedicat per fer cada pràctica

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

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

L'avaluació de la pràctica serà sobre 10 punts, 8 punts o 6 punts, en funció de quan es lliuri la pràctica (i sigui acceptada):

Sobre 10 punts: fins al 21 de maig del 2023, a les 23:59h.

Sobre 8 punts: des del 22 de maig al 4 de juny del 2023, a les 23:59h.

Sobre 6 punts: des del 5 de juny del 2023 al 2 de juliol del 2023, a les 23:59h.