Pràctica final - Algorísmia - 2020
Versió: 1.0

Pràctica final

A continuació es defineix l'enunciat de la pràctica final de l'assignatura d'algorísmia de segon d'enginyeria informàtica.

La primera secció introdueix el contingut de la pràctica, la segona els elements que ha de contenir l'entrega final i la tercera els detalls del que s'ha d'implementar.

Noteu que a la introducció la puntuació final de la pràctica només suma 8 punts, els dos punts restants es calcularan amb la professionalitat de la solució aportada.

Enunciat

El propietari de l'aplicació d'escacs no està molt content amb la seva aplicació, diu que no compleix amb les expectatives del que esperava. Amb la solució que li hem proposat, que tenia un bug col·locant les peces a la primera fila i columna, no creu que nosaltres siguem capaços de desenvolupar tota la lògica de negoci de l'aplicació d'escacs.

Per a continuar treballant amb nosaltres ens ha proposat un repte. Aquest repte consisteix en implementar, i integrar, la solució a tres problemes a dins de la seva aplicació. Aquests problemes estan descrits a continuació.

Problema 1: Col·locació de n reines a sobre d'un tauler nxn (1 punts)

Considera el problema de situar n reines en un tauler de n×n caselles de manera que cap de les reines es puguin matar. El moviments de les reines són els mateixos que al joc d'escacs.

Problema 2: Les peces que no es maten (6 punts)

Col·locar m peces d'escacs a un tauler de dimensió nxn de forma que no es matin entre elles. S'han d'incorporar dues peces inventades pels autors.

Problema 3: Els moviments del cavall (1 punt)

Implementeu un programa backtracking al problema dels moviments del cavall, dissenyant un programa amb una interfície gràfica agradable en la que es posi de relleu la ruta seguida pel cavall.

Entregables

Entregable 1 (Implementació problema 1):

Funcionalitat que col·loqui n reines sobre un tauler de n×n caselles de manera que cap de les reines es pugui matar.

Els usuaris seleccionaran manualment la configuració del problema:

- La mida del tauler (n).
- La posició a on es situarà la primera reina.

Entregable 2 (Implementació problema 2):

Funcionalitat que col·locar m peces de diferent tipus a un tauler de dimensió nxn, de forma que no es matin entre elles

Els usuaris seleccionaran manualment:

- La mida del tauler (n).
- Un llistat de les peces que es volen col·locar a sobre del tauler.

Entregable 3 (Implementació problema 3):

Funcionalitat que calculi els moviments que ha de realitzar un cavall per a moure's per a visitar totes les caselles del tauler únicament una vegada.

Els usuaris seleccionaran manualment:

- La mida del tauler (n).
- La posició inicial del cavall.

Entregable 4 (Memòria):

Memòria de la pràctica amb els següents punts principals:

1. Jerarquia de classes (Diagrama UML).
2. Disseny dels problemes (esquema recursiu).
3. Solucions adoptades (Optimitzacions, ...).
4. Cost computacional dels algorismes implementats.
5. Manual d'usuari.

Detalls

Els resultats als tres problemes s'han de mostrar en una GUI.

Tota la implementació s'ha de dur a terme mitjançant el paradigma de disseny de programació orientada a objectes.

Problema 1

Pel problema de les n reines es dóna total llibertat en el seu desenvolupament. El problema de les n reines és un problema conegut, trobareu informació suficient a la xarxa: https://es.wikipedia.org/wiki/Problema_de_las_ocho_reinas .

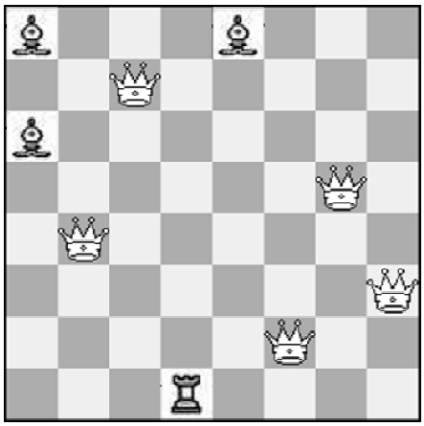
Recordeu que en un tauler nxn el nombre de reines que s'han col·locar és n. La GUI ha de permetre seleccionar el valor de n i la posició de la primera reina.

Problema 2

Per aquest segon problema es demana que s'implementi una GUI que permeti seleccionar el nombre de caselles d'un tauler nxn i un conjunt de m peces d'escacs.

Per exemple, si l'usuari vol col·locar cinc reines, tres alfils i una torre en un tauler de 12x12, llavors m=9 y n=12.

La següent figura mostra un exemple dels resultats que mostrarà l'aplicació per pantalla amb la configuració anterior, fixeu-vos en que cap de les peces del tauler mata a cap altre.



Requisits que s'han de complir:

1. Ha de disposar de les peces següents: reina, rei, cavall, alfil y torre, a més de dues peces inventades pel desenvolupador (o grup de desenvolupadors).

2. La solució que donarà l'aplicació final enfront d'un determinat problema ha de ser calculada per un algoritme de backtracking.

3. El programa ha de disposar d'una interfície gràfica d'usuari a on es puguin formular els problemes i veure la solució final obtinguda per l'aplicació, si existeix. Si no hi ha solució enfront d'un determinat problema s'haurà d'informar a l'usuari d'aquesta conclusió.

Problema 3:

El problema dels moviments del cavall consisteix en calcular els moviments que ha de fer un cavall per a visitar totes les caselles del tauler, visitant cada casella una única vegada.

La solució es mostrarà en un tauler de n×n caselles, de manera que a cada casella s'hi inclogui el nombre de moviments que ha fet el cavall per arribar-hi. L'usuari decidirà la mida del tauler (n) i la posició inicial del cavall.

Criteris aprovat:

1. El programa haurà de poder ser compilat i executat sense cap tipus de problema.
2. Els resultats es mostren mitjançant una GUI.
3. L'autoria de la pràctica ha d'estar demostrada. En cas de copiar codi, agafar idees de la xarxa, col·laboració amb companys d'altres grups, ... ha de quedar degudament documentat. Recordeu que, en aquests casos, heu de saber el que esteu fent, en cas contrari es considera suspès.