

Inlämningsuppgift 2: File Crawler

Introduktion och syfte

Målet är att sätta sig in i hur vi kan hantera filer och mappar med hjälp av rekursion. Första bonusuppgiften lär er använda hash tabell i praktiken.

Uppgift

Uppgiften går ut på att skriva ett simpelt lokalt sökprogram som tar in en textsträng från användaren och gör en sökning i katalogen där programmet körs, samt dess underkataloger.

Programmet ska gå igenom innehållet i alla filer för att se om den angivna textsträngen finns i filen.

- Programmet ska starta med ett meddelande som säger åt användaren att skriva in ett sökord.
- Om programmet hittar en fil som innehåller textsträngen så ska filens absoluta sökväg skrivas ut på en egen rad.
- Om filen inte innehåller textsträngen så ska ingenting skrivas ut för den filen.
- Om filen (eller en katalog) inte går att läsa så ska ett felmeddelande som minst innehåller filens/katalogens absoluta sökväg skrivas ut.

Tips

Skapa en ny mapp/projekt för inlämningen och lägg in filerna från "Assignment 3" i lärarens repo.

Filsökning kan vara enklare att få till i Python än C++. I python får ni skapa programmet själva från grunden (ni vill använda biblioteket "os"). I C++ kommer jag lägga upp en grundstruktur med CMake för att använda Boosts filesystem bibliotek.

Det finns andra möjlighet att läsa filer och filstruktur i C++ som ni kan använda:

- `fstream` (C++ STL) (`ifstream`, `ofstream`)
- `C stdio.h`: `fopen`, `fread`, `fwrite`, `fclose`

Bonusuppgifter

Generellt: Implementation i Python och C++, upp till 2 bonuspoäng för alla inlämningsuppgifter. Om ni löser denna uppgift i både Python och C++, räcker det att minst en använder rekursion.

Bonusuppgift 1: (1.5 bonuspoäng)

Gör en alternativ implementering av uppgiften där ni har en funktion som indexerar hela katalogen.

Denna indexing ska sedan kunna användas för att hitta ord snabbt:

- Indexeringsfunktionen ska gå igenom hela katalogstrukturen och alla filer
- För varje fil ska den indexera varje ord (dela upp filens text i ord) och spara i vilken fil ordet kan hittas i hash tabellen (ordet som key och fil-sökvägar som value)
- Tänk på att samma ord kan finnas i flera filer. Värdena i hashtabellen måste alltså rymma flera filsökvägar
- Hash table är en abstract data typ. Vilken faktisk datatyp ni ska använda beror på programmeringsspråk.
- Main funktionen ska först kalla på indexeringsfunktionen. Sedan den vanliga sök-funktionen. Sök-funktionen ska använda hash-tabellen för att hitta och skriva ut sökvägen till alla filer där ordet förekommer.

Observera: Ni måste göra en implementation/funktion som följer grunduppgiften också. Dessa kan vara i samma eller olika programmeringsspråk.

Bonusuppgift 2 (Upp till 1.5 bonuspoäng)

Inkludera argument till ditt program, som du väljer från listan nedan. Varje implementerat argument ger 0.5 bonuspoäng (upp till 3 argument). (Har ni någon egen idé på argument ni vill implementera kan det vara ok, fråga mig). Själva söksträngen kan också vara en option om ni vill eller hanteras som user input. Bonusuppgift 2 kan vara en påbyggnad av bonusuppgift 1 eller bygga på grunduppgiften.

Exempel:

- Search Mode: Implement options to perform a case-sensitive or case-insensitive search.

Argumenten till programmet kan då se ut som följer:

```
$$$  
$ file_crawler.exe -c # Performs case sensitive search  
$ file_crawler.exe -i # Performs case insensitive search  
$$$
```

Eller i python:

```
$$$  
$ python file_crawler.py -c # Performs case sensitive search  
$ python file_crawler.py -i # Performs case insensitive search  
$$$
```

Övriga förslag på argument / options:

- File Types: Allow the user to specify the type of files to be searched (e.g., .txt, .pdf, .docx, md).
\$ file_crawler.exe --file-types=txt,md
- Output File: Provide an option to write the results to a file, instead of printing to the console.
- Display line numbers: Also print line numbers
- Search Depth: Set a limit on the number of subdirectories to be searched, so that the search doesn't go too deep.
- Search in Sub-directories only
- Options for recursive/non-recursive: Set if the search shall be recursive or not (non-recursive: Only search top folder)
- Delimiter: (If using index/hashtable). Allow user to specify delimiter for words to use
- Exclude Directories: Allow the user to specify directories to be excluded from the search.

Redovisning

Källkoden ska vara pushad till ett eget repository på GitHub.

Inlämningen sker genom att en länk till gitrepo skickas in i Omniway.