

Párhuzamos és elosztott rendszerek programozása féléves feladat dokumentáció Farkas Norbert - J0W4RG

A féléves feladatom során készíték egy zip fájl jelszó feltörő programot, brute-force módszert alkalmazva a jelszó keresést fogom párhuzamosítani. A DotNetZip library segítségével fogom kezelni a a tömörített fájlt.

A program elindítása után lehetőség lesz megadni egy zip fájlt, aminek a jelszavát szeretnénk feltörni, majd megadhatunk egy minimális illetve maximális jelszóhosszt is, hogy hány karakter hosszú jelszótól kezdjük a vizsgálatot és meddig menjünk. Alapértelmezetten, ha nem adunk meg jelszóhosszt a program egy karakter jelszóhosszával fog indulni és addig megy amíg meg nem találja a jelszót, de meg is állíthatjuk a keresést. A jelszó keresést szűkíthetjük bizonyos típusú karakterekre is (kis és nagybetűk, számok, stb.), ezzel csökkentve a kombinációk számát és gyorsítva a keresést, ha tudjuk, hogy milyen karakterek nem szerepelnek a jelszóban.

A beállítások után indulhat a keresés. Első lépésben megvizsgálom a számítógépben található processzormagok számát, majd ezzel megegyező számú taskot indítok a kereséshez. A keresés akkor áll le, ha megtaláltuk a jelszót, vagy megadtunk egy maximális jelszóhosszt, és végig mentünk minden kombináción, de nem találtuk meg a jelszót.

Az előre beállított karakterlistát annyi részre osztom, ahány taskot indítok. Mivel minden task ugyanazt a feladatot fogja elvégezni csak más karakterekkel, így körülbelül egyszerre fognak végezni, kivéve ha a karakterlista elosztása során egy task, több kezdőkaraktert kap. Például, ha 2 taskot indítok és karakterlistának csak számokat adtunk meg, akkor az 1.task fog felelni a 0-4-el kezdődő kombinációkért, a 2.task az 5-9-el

kezdődőekért. Ha nem állítottunk be minimális jelszóhosszt, akkor 0-tól 9-ig végigpróbáljuk a számokat, majd lépünk a két karakter hosszú kombinációkhoz, ugyanolyan elven mint az egy karakteresnél. Például, ha négy karakter hosszúságnál járunk, akkor az 1. task az első három karaktert beállítja 000-ra majd a negyedik karaktert 0-tól 9-ig végigpróbálja, ezután a harmadik karaktert egyesre állítja és a negyedik karaktert megint végigpróbálja stb. Amikor elérkezik a rá kiosztott karakterlista végére, ez esetben 4999, akkor elkezdi vizsgálni az öt karakter hosszú kombinációkat is, ha a megadott maximális jelszóhossz alatt vagyunk még.

A program futása közben minden szárról megjelenítek információkat az aktuális vizsgált kombinációról. Ha megtaláltuk a jelszót, akkor az megjelenítem, ha megállítottuk a programot, akkor megjelenítem az utoljára teljesen átvizsgált jelszóhosszt, hogy ha újra indítjuk a keresést, akkor megadhatjuk azt minimális jelszóhossznak.

A programot elkészítem szekvenciálisan is, hogy össze lehessen mérni a párhuzamosított megoldás gyorsulásának mértékét a szekvenciálishoz képest. A szekvenciális program is ugyanezen az elven fog működni, csak itt egy szálon sorban megyünk végig a karakterkombinációkon.