Teljesítményjellemzők vizsgálata

Szoftverfejlesztés laboratórium 2

Mérési útmutató

Készítette: Hajdu Ákos, Barta Patrik, Vörös András

Verzió: 2.0

2018.

Budapesti Műszaki és Gazdaságtudományi Egyetem

Méréstechnika és Információs Rendszerek Tanszék

# Előkészületek

## Általános tudnivalók

A mérési feladatokat és az elvégzésükhöz szükséges, a kiadott segédlet által nem tartalmazott információkat jelen útmutató tartalmazza. A feladatok dokumentálását jelen útmutató kijelölt helyeire végezzék! A feladatoknál mindig jelöljük, hogy kell-e dokumentálni, és ha igen, akkor milyen formában (táblázat, diagram, képernyőkép, forráskód, magyarázat).

**Fontos:** A feladatok megoldása során az eredmények értékelése is a feladat része: az eredmény olyan-e, amit vártunk, reális-e az az eredmény, stb. Ezeknek a végiggondolása a mérés része, szándékosan hiányoznak a „Most azt kellene látni, hogy…” részek!

## Szoftverkörnyezet

A mérés elvégzéséhez a következő szoftverelemek állnak rendelkezésre a virtuális gépen. A gépen minden szükséges fájl megtalálható a **C:\ \_profiling** mappában!

A mérés elvégzéséhez szükséges szoftverek:

* Microsoft Visual Studio 2013 Ultimate (Update 5)
* Windows Performance Recorder (WPR)
* Windows Performance Analyzer (WPA)
* Microsoft Office (Word / Excel)

### Bemeneti állományok

A bemeneti állományok a C:\\_profiling mappában találhatóak. Ha a feladat nem írja elő másképp, akkor ezekből lehet dolgozni.

* \_src: a mérés során használt alkalmazások forráskódja (Visual Studio projektek)
* 1.5.1 Eredmények: előre felvett profiling eredmények
* Disk IO: előre felvett diszk IO adatok
* Teszt képek: az alkalmazás futtatása során felhasználható teszt képek

# Profiling

## Ismerkedés az alkalmazással

**Indítsa el a DemoApp.Profiling alkalmazást és próbálja ki a működését a bemeneti állományokból válogatva (Teszt képek mappa)! Nézzen bele az alkalmazás forráskódjába (beleértve a DemoApp.Filters projektet is)! Az effektek működését nem szükséges végignézni.**

*Megjegyzés: A mintaalkalmazás csak oktatási célokat szolgál, valós környezetben a hosszú ideig tartó műveleteket nem érdemes a grafikus felülethez tartozó szálon futtatni.*

Javasolt időráfordítás: maximum 5 perc.

**A feladatot nem szükséges dokumentálni.**

## Mintavételezett profiling

A következő feladatok során mintavételezett profilingot kell végrehajtani és az eredményét értékelni. A feladatok során célszerű „Release” módban futtatni a programot, mert úgy gyorsabb.

***1.2.1. Lépéssorozat futtatása mintavételezett profilinggal***

**Indítson el egy mintavételezett profilingot és hajtsa végre a következő lépéseket:**

**- Nyissa meg valamelyik képet.**

**- Alkalmazza rá az olajfestmény effektet.**

**- Zárja be az alkalmazást.**

Javasolt időráfordítás: 5 perc.

**A feladatot nem szükséges dokumentálni.**

***1.2.2. Ismerkedés az eredménnyel***

**Ismerkedjen meg a mérési eredményeket megjelenítő nézetekkel! Az áttekintő nézetben próbálja ki a “Show All Code” opciót! A hívási fa nézetben próbálja ki a “Noise Reduction” funkciót!**

Javasolt időráfordítás: 10 perc.

**A feladat dokumentálandó (képernyőképek).**

Megoldás:

***1.2.3. Hot Path***

**Írja le röviden a Hot Path definícióját! Jelen esetben mi a Hot Path?**

Javasolt időráfordítás: 5 perc.

**A feladat dokumentálandó (képernyőképek, magyarázat).**

Megoldás:

## Instrumentált profiling

A következő feladatok során instrumentált profilingot kell végrehajtani és az eredményét értékelni.

***1.3.1. Lépéssorozat futtatása instrumentált profilinggal***

**Indítson el egy instrumentált profilingot és hajtsa végre a következő lépéseket:**

**- Nyissa meg valamelyik képet.**

**- Alkalmazza rá az olajfestmény effektet.**

**- Zárja be az alkalmazást.**

Javasolt időráfordítás: 5 perc.

**A feladatot nem szükséges dokumentálni.**

***1.3.2. Ismerkedés az eredménnyel***

**Tekintse át a mérési eredményeket! Milyen különbségek figyelhetőek meg a mintavételezett és az instrumentált nézetek között? Ugyanaz a mért adatok mértékegysége?**

Javasolt időráfordítás: 10 perc.

**A feladat dokumentálandó (képernyőképek, magyarázat).**

Megoldás:

## Memóriaszivárgás keresése

***1.4.1. Memóriaszivárgás keresése***

**A DemoApp.Profiling alkalmazásban szándékosan elrejtettünk egy potenciális lehetőséget a memóriaszivárgásra. Indítsa el a Visual Studio „Memory Usage” eszközét erre a projektre. Nyisson meg egymás után több képet (legalább kettőt) és figyelje a memóriafogyasztást. Hajtson végre egy szemétgyűjtést (Force GC) miután megnyitott több képet. Mivel magyarázható a jelenség? *Segítség: Mi okozhat felügyelt környezetben memóriaszivárgást?***

Javasolt időráfordítás: 10 perc.

**A feladat dokumentálandó (képernyőképek, magyarázat).**

Megoldás:

***1.4.2. Memóriaszivárgás megszüntetése (nem kötelező feladat, IMSc pontért megoldható)***

**Keresse meg a memóriaszivárgás pontos okát és javítsa ki! Futtassa le újra a „Memory Usage” eszközt, hogy meggyőződjön a javítás sikereségéről!**

Javasolt időráfordítás: 10 perc.

**A feladat dokumentálandó (képernyőképek, forráskód részlet, magyarázat).**

Megoldás:

## Mérési adatok elemzése

***1.5.1. Adatok letöltése***

**Egy kép különböző méretű változatain előre elvégeztünk méréseket (1.5.1 Eredmények mappa). A CSV fájlban megtalálhatóak a képek paraméterei (fájlnév, szélesség, magasság, pixelszám, fájlméret) illetve az olajfestmény effekt általunk mért futási ideje.**

**Tekintse át a képeket és a mérési eredményeket!**

Javasolt időráfordítás: maximum 5 perc.

**A feladatot nem szükséges dokumentálni.**

***1.5.2. Futási idő a pixelszám függvényében***

**Ábrázolja az effekt futási idejét a pixelszám függvényében! Ügyeljen a megfelelő diagramtípus megválasztására! Milyen függvénnyel írható le a pixelszám és a futási idő közötti összefüggés?**

Javasolt időráfordítás: 15 perc.

**A feladat dokumentálandó (diagram, magyarázat).**

Megoldás:

***1.5.3. Futási idő a szélesség/magasság függvényében***

**Ábrázolja az effekt futási idejét a kép szélességének vagy magasságának függvényében. Mi okozza a kiugró értéket? Milyen függvénnyel írható le a szélesség/magasság és a futási idő közötti összefüggés?**

Javasolt időráfordítás: 10 perc.

**A feladat dokumentálandó (diagram, magyarázat).**

Megoldás:

***1.5.4. Futási idő a fájlméret függvényében (nem kötelező feladat, IMSc pontért megoldható)***

**Ábrázolja az effekt futási idejét a fájlméret függvényében. Mi okozza a kiugró értéket? Milyen függvénnyel írható le a fájlméret és a futási idő közötti összefüggés?**

Javasolt időráfordítás: 10 perc.

**A feladat dokumentálandó (diagram, magyarázat).**

Megoldás:

# Nyomkövetés

A nyomkövetéshez nincs szükség Visual Studio-ra, ezért a teljesítmény javítása érdekében érdemes a programot lefordítani (Release módban), majd Visual Studio-t bezárva a projekt „bin/Release” mappájából futtatni.

## Nyomkövetés rögzítése

***2.1.1. Rögzítés***

**Indítson nyomkövetést a WPR segítségével, minden beépített “Resource Analysis” profilt bekapcsolva! A tárolt nyomot állományba rögzítse (Logging mode)! A nyomkövetés során az alábbi lépéseket hajtsa végre:**

**Indítsa el a DemoApp.Profiling alkalmazást!**

**- Nyisson meg egy képet!**

**- Alkalmazza rá az oljafestmény effektet!**

**- Nyisson meg egy másik képet!**

**- Alkalmazza rá az oljafestmény effektet!**

Javasolt időráfordítás: 5 perc.

**A feladatot nem szükséges dokumentálni.**

## Eredmények megtekintése WPA-ban

***2.2.1. Ismerkedés a WPA-val***

**Indítsa el a Windows Performance Analyzer eszközt és töltse be az előző lépésben rögzített nyomot!
Tekintse át a felület képességeit!**

Javasolt maximális időráfordítás: 10 perc.

**A feladatot nem szükséges dokumentálni.**

***2.2.2. Viselkedés felderítése, CPU***

**Konfiguráljon fel CPU Usage (Sampled) nézetet, amely az alkalmazás CPU használatát mutatja grafikusan!**

**Konfigurálja be úgy a nézetet, hogy a CPU használat táblázatos megjelenítése is látható legyen!**

**A nézetet állítsa be úgy, hogy az alkalmazás által használt modulokat és függvényeket is mutassa!**

**Állítson be szűrőt, hogy csak a releváns (az általunk mért alkalmazáshoz tartozó) adatok látszódjanak!**

**Grafikusan azonosítsa az effekt meghívását a teljes CPU használatban kijelöléssel!**

**Képernyőképpel támassza alá, amin a táblázatos nézetben látszik a kijelölt függvény (oldja fel a szimbólumokat), illetve ennek hatására a grafikonon a CPU használat! A szimbólumok feloldása előtt érdemes beállítani, hogy csak a mi alkalmazásunk könyvtárában keressen (lásd a segédletet).**

Javasolt időráfordítás: 20 perc.

**A feladat dokumentálandó (képernyőképek, magyarázat).**

Megoldás:

***2.2.3. Viselkedés felderítése, diszk és fájl I/O***

**A virtuális gépen megtalálható egy előre felvett nyom (Disk IO könyvtárban). Azt rögzítettük, ahogy a FiltersDemo.exe nevű alkalmazással megnyitottunk néhány képet. A feladat kideríteni a diszk és fájl I/O nézetek segítségével, hogy melyik képet hányszor nyitottuk meg.**

**Konfiguráljon fel egy nézetet, amely a rendszer diszkhasználatát (Disk Usage) és fájlműveleteit (File I/O) mutatja grafikusan! Konfigurálja be úgy a nézeteket, hogy a táblázatos megjelenítések is láthatóak legyenek!**

**Állítsa be, hogy mindkét nézetben láthatóak legyenek az alkalmazás által használt állományok is! (A “Disk Usage” nézet szerkesztőjében ehhez a “Path Name” attribútumot kell hozzáadni, a “File I/O” nézetben a “File Name” jelöli a fájl nevét.)**

**Mutassa meg a File I/O nézetben, hogy mely képeket olvasta be az alkalmazás (Read esemény)!**

**Mely fájloknál nyúlt az alkalmazás a merevlemezhez (Disk I/O nézet)?**

**Ugyanazt mutatja a két grafikon? Mivel magyarázható a jelenség?**

**Az egyik képet kétszer nyitottuk meg. A fájl I/O időket vizsgálva mit tapasztalunk? Mivel magyarázható a jelenség?**

Javasolt időráfordítás: 25 perc.

**A feladat dokumentálandó (képernyőképek, magyarázat).**

Megoldás:

# Párhuzamosítás

## Gép paraméterei

***3.1.1. Gép paramétereinek dokumentálása***

**Dokumentálja a gép alapvető teljesítmény paramétereit (processzormagok száma, órajel, RAM, stb.) amin dolgozik! Amennyiben a saját gépe több processzormaggal rendelkezik, mint a virtuális gép és van .NET keretrendszere, dolgozzon azon!**

Javasolt időráfordítás: 5 perc.

**A feladatot dokumentálandó (táblázat).**

Megoldás:

|  |  |
| --- | --- |
| **Paraméter** | **Érték** |
| Paraméter1 | Érték1 |
| Paraméter2 | Érték2 |
| … | … |

##  Gyorsulás mérése

***3.2.1. Az alkalmazást felkészíteni a mérésre***

**Nyissa meg a DemoApp.Parallelization projektet! Ismerkedjen a forráskóddal és futtassa a programot! Készítse fel az alkalmazást, hogy a futásideje mérhető legyen!**

**(Ezt lehet akár profilozással, akár a .NET-es Stopwatch-csal, vagy bármilyen – megfelelő pontosságú – saját megoldással is.)**

Javasolt időráfordítás: 15 perc.

**A feladat dokumentálandó (forráskód, magyarázat).**

Megoldás:

***3.2.2. Mérés elvégzése és eredmények értékelése***

**Végezze el a futásidő mérését úgy, hogy a párhuzamos szálak maximális száma (max. degree of parallelism) rendre 1, 2, 4, 8 legyen!**

**Az eredményeket táblázatosan és grafikusan is dokumentálja!**

**Röviden értékelje az eredményeket! Megfeleltek a gép paraméterei alapján az előzetes várakozásoknak?**

Javasolt időráfordítás: 25 perc.

**A feladat dokumentálandó (táblázat, diagram, magyarázat).**

Megoldás:

##  Az alkalmazásban lévő soros rész párhuzamosítása

***3.3.1. Párhuzamosítás (nem kötelező feladat, IMSc pontért megoldható)***

**Az alkalmazás úgy lett megírva, hogy a soros rész (képek betöltése) is párhuzamosítható lenne.**

**Adjon valamilyen megoldást a párhuzamosításra (pl. szálak, TPL, …) és végezze el újra a mérést!**

**A megírt kódot és az eredményeket is dokumentálja, értékelje!**

Javasolt időráfordítás: 25 perc.

**A feladat dokumentálandó (forráskód, táblázat, diagram, magyarázat).**

Megoldás: