



Informatikai technológiák laboratórium 2

Teljesítményjellemzők vizsgálata

Mérési útmutató

Készítette: Kocsis Imre, Izsó Benedek, Vörös András

ikocsis@mit.bme.hu

2013.

Verzió: 1.11

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 segédlet és az útmutató mellé biztosítunk még egy jegyzőkönyv-sablont is.

Kérjük, hogy a feladatok dokumentálásához ezt a jegyzőkönyv-sablont használják.

A mérés során egy Windows 7 virtuális gépen fognak dolgozni.

- A virtuális laborban (VCL) érhető el a gép, „ITLab2 MIT2” néven.
- Felhasználónév/jelszó: amit a VCL környezet adott. Fontos ezt használni, mivel ha ez a felhasználó nem lép be, megszűnik a foglalás, és minden adat elveszik.
- A foglalás lejártakor is elvesznek az adatok, ezért fontos a **gyakori mentés**, és a **fájlok saját gépre másolása** (vagy jegyzőkönyv saját gépen készítése).

Szoftverkörnyezet

A mérés elvégzéséhez a következő szoftverelemek állnak rendelkezésre a virtuális gépen.

AForge.NET

Az AForge.NET¹ egy gépi látás és mesterséges intelligencia funkciókat megvalósító C# keretrendszer. A keretrendszer egy példánya rendelkezésre áll a

```
C:\code\AForge.NET-2.2.4
```

elérési úton. A mérés során több feladatban is használt FiltersDemo mintaalkalmazás elérési útja ezen belül:

```
C:\code\AForge.NET-2.2.4\Samples\Imaging\FiltersDemo
```

A mintaalkalmazáshoz tartozó Visual Studio megoldásállomány (.sln – „solution” – állomány) ebben a mappában található. Az előre lefordított, **közvetlenül is indítható** mintaalkalmazás:

```
C:\code\AForge.NET-2.2.4\Samples\Imaging\FiltersDemo\bin\Debug\FiltersDemo.exe
```

Bemenetként használható képek

A mintaalkalmazás bemenetként használni javasolt képeket tároló mappa:

```
C:\inputs
```

A képek forrása:

- steve1 mappa: a Wired.com Flickr fotófolyama². Licenz: Creative Commons Attribution-NonCommercial 2.0 Generic
- steve2 mappa: a Wired.com Flickr fotófolyama³. Licenz: Creative Commons Attribution-NonCommercial 2.0 Generic

¹ <http://www.aforgenet.com/>

² <http://www.flickr.com/photos/wiredphotostream/6674308877/>

- bzmot mappa: a Wikipedia Burrows nevű felhasználójának saját felvétele⁴. Licenz: Creative Commons Attribution 3.0 Unported

Visual Studio

A virtuális gépre telepítve van a Microsoft Visual Studio 2012 Ultimate. Az eszköz elindítható az asztalon található gyorsindító-gombbal és a Start menüből is.

Windows Performance Recorder (WPR)

A virtuális gépen telepítve van a Windows Performance Recorder (WPR). Az eszköz elindítható az asztalon található gyorsindító-gombbal és a Start menüből is.

Windows Performance Analyzer (WPA)

A virtuális gépen telepítve van a Windows Performance Analyzer (WPA). Az eszköz elindítható az asztalon található gyorsindító-gombbal és a Start menüből is.

További eszközök

Kényelmi szolgáltatásként a virtuális gépen megtalálhatóak még a további alkalmazások is, melyek a Tálcáról is indíthatóak:

- Notepad++
- Google Chrome
- Microsoft Office 2010 Pro: Word és Excel

1 Ismerkedés a FiltersDemo alkalmazással

I/1. Feladat. Indítsa el a FiltersDemo alkalmazást és próbálja ki működését a megadott bemeneti állományokból válogatva!

Javasolt időráfordítás: maximum 10 perc.

A feladatot nem szükséges dokumentálni.

I/2. feladat. Egyszerű állapotgéppel⁵ modellezze az alkalmazás a felhasználó szemszögéből nézve érzékelhető különböző állapotait!

Ügyeljen arra, hogy explicit megjelenjenek az alkalmazás azon állapotai, amikor emberi interakciót nem fogad, hanem a „háttérben dolgozik”! A különböző szűrőket (filter) nem szükséges megkülönböztetnie. (5 állapotnál többet nem javasolt felvenni.)

Javasolt időráfordítás: ~10 perc.

A feladat rövid érveléssel alátámasztva dokumentálandó.

³ <http://www.flickr.com/photos/wiredphotostream/6302995233/>

⁴ http://hu.wikipedia.org/wiki/F%C3%A1jl:Bzmot_Mak%C3%B3-%C3%A1jv%C3%A1ros_02.JPG

⁵ Ismétlésképpen: http://en.wikipedia.org/wiki/Finite-state_machine

I/3. feladat. Indítsa el a Visual Studio 2012 Ultimate-et és ismerkedjen meg az alkalmazás forráskódjával!

Javasolt maximális időráfordítás: 10 perc.

A feladatot nem szükséges dokumentálni.

I/4. Feladat. Ha eddig nem tette, futtassa valamelyik nagyobb számítási komplexitású szűrőt (pl. az *Oil Painting*-et) az egyik nagyobb méretű képen (pl. a *Bzmot-on*)!

A forráskód ismeretében miért nem fogad az alkalmazás felhasználói interakciókat számítás közben?

Röviden ismertesse, hogy hogyan módosítaná az alkalmazást e hátrányos tulajdonság orvoslásához! (A módosítást nem kell elvégezni.)

Javasolt maximális időráfordítás: 10 perc.

A feladat dokumentálandó.

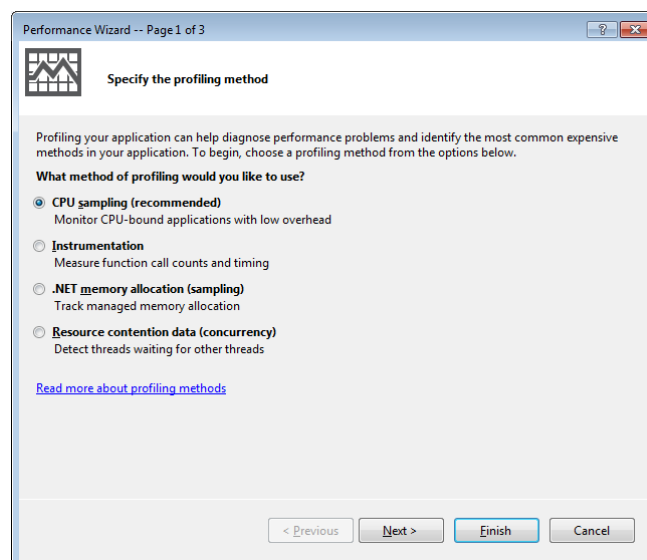
2 Profiling

2.1 Mintavételezés

II/1. Feladat. Tekintse át a „teljesítmény-varázsló” által felkínált lehetőségeket (ANALYZE → Launch Performance Wizard) és hajtson végre egy mintavételezett profiling-ot! Ismerkedjen meg a mérési eredményeket megjelenítő nézetekkel!

Javasolt maximális időráfordítás: maximum 15 perc.

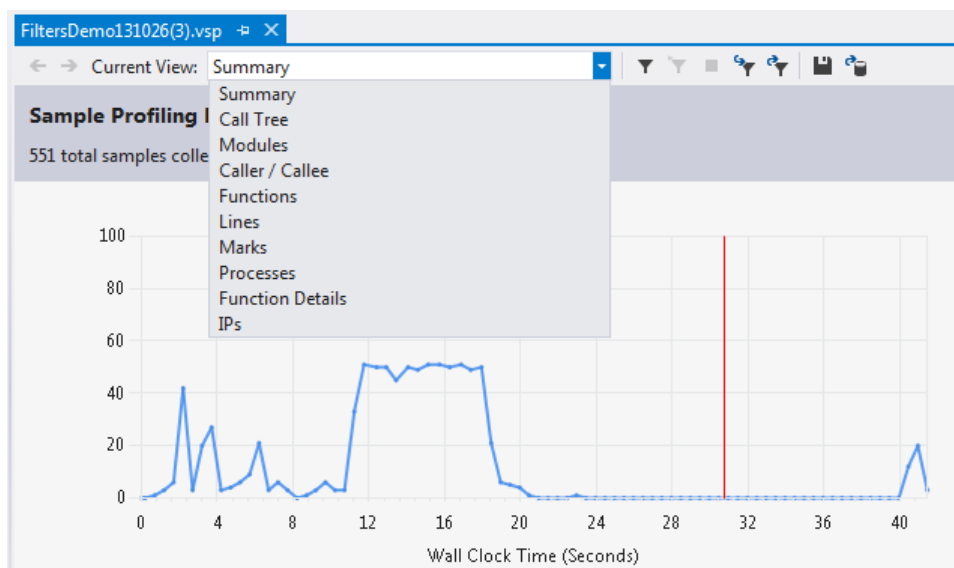
A feladatot nem szükséges dokumentálni.



1. ábra CPU mintavételezés beállítása

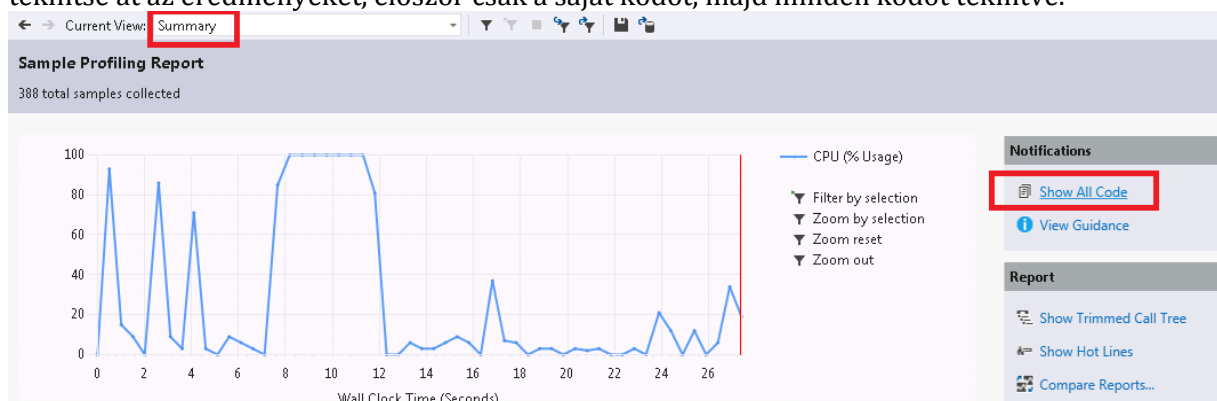
A következő munkamódszert javasoljuk:

1. A nézeteknél sorrendben haladjon (Summary → ... → IPs)



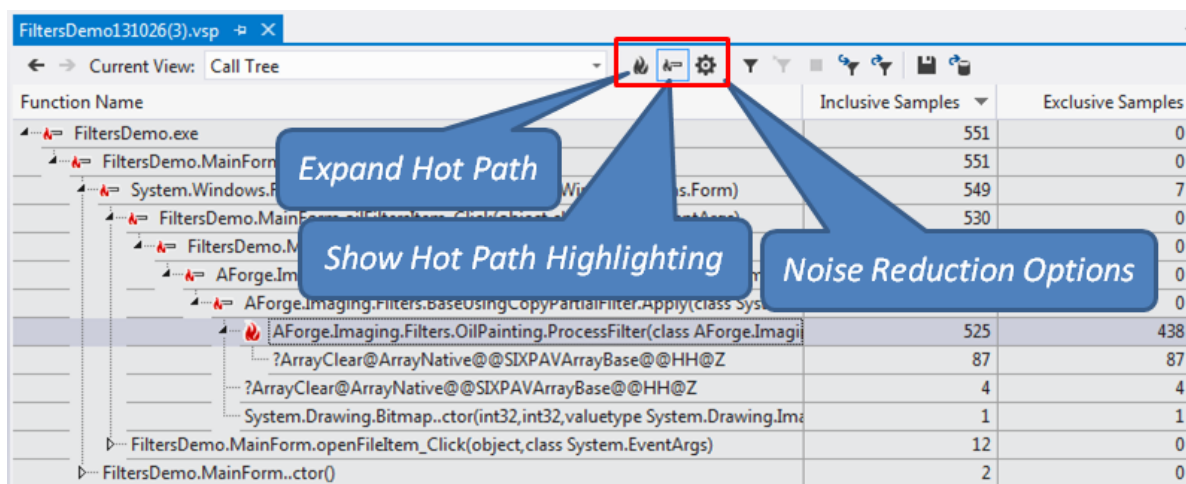
2. ábra Elérhető nézetek

2. Az áttekintő nézetben először ne kapcsolja be a „Show All Code” opciót; inkább két menetben tekintse át az eredményeket, először csak a saját kódot, majd minden kódot tekintve.



3. ábra Show All Code és Just My Code közötti váltás

3. A hívási fánál mindenképp próbálja ki a „Hot Path” és „Noise Reduction” funkciókat!



4. ábra Hot Path és Noise Reduction opciók

4. A nézetek áttekintése után ismerkedjen meg a „Performance Explorer”-ben megjelent elemek kontextusmenüjével is!

II/2. Feladat. Hajtson végre egy mintavételezett profiling-ot a `steve1` mappában található képen, az `Oil Painting`, a `Jitter` és a `Sobel edge detector` szűrőket alkalmazva! A következő feladatok az így előálló teljesítmény-jelentésre fognak vonatkozni.

Javasolt maximális időráfordítás: pár perc.

A feladatot nem szükséges dokumentálni.

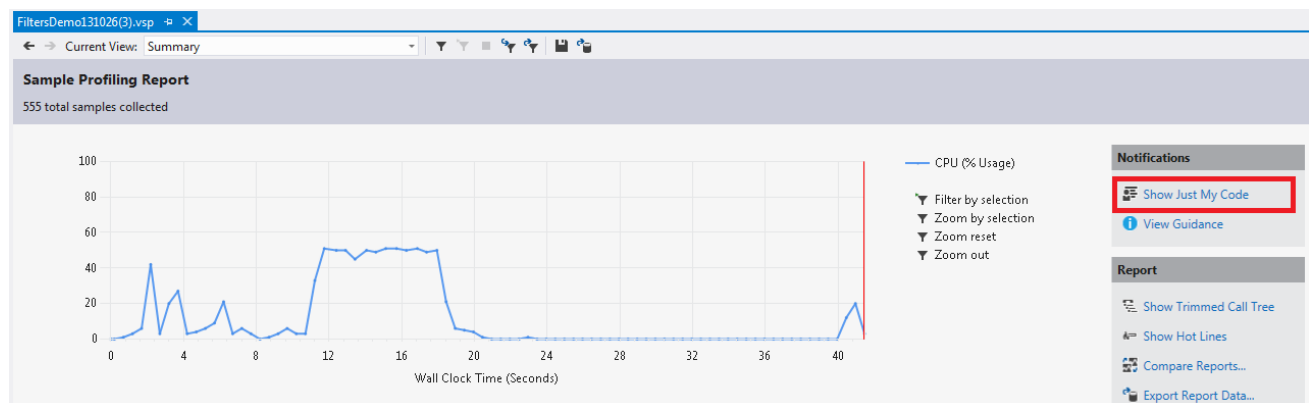
II/3. Feladat.

Mi az alkalmazás „Hot Path”-a?

Exkluzív és inkluzív minták tekintetében mit mond el az alkalmazás teljesítményviszonyairól? (Az analízist „Just My Code” módban végezzük!)

Javasolt maximális időráfordítás: 10 perc.

A feladat dokumentálandó.



5. ábra Analízis futtatása „Just My Code” módban

II/4. Feladat.

A mérés alapján az `Oil Painting` szűrő (`OilPainging.ProcessFilter()` függvény) „aktív” ideje (CPU-mintavételezett értelemben) milyen faktoroktól függ? (Például ilyen a kép mérete, ezen kívül még írjon kettőt!)

Milyen feladata lehet az általa hívott függvényeknek? (Mutasson is egy ilyen függvényt a mérési adatok alapján!)

Mit feltételezhetünk a hívott függvények futási hatékonyságáról?

Javasolt maximális időráfordítás: maximum 10 perc.

A feladat dokumentálandó.

II/5. Feladat.

Írja le, mi a *speedup* (gyorsítás) fogalma! Nézzen utána a Wikipedián! Hogyan lehet kifejezni egy párhuzamos algoritmus futási idejét a *speedup* függvényében, ha ismert a szekvenciális algoritmus futási ideje.

Tegyük fel, hogy az Oil Painting szűrő (tehát az előzőekben megfigyelt függvény) hatékonyságát úgy tudjuk növelni, hogy i) az általa végzett további hívások száma/gyakorisága nem változik, de ii) a kódot sikerül felgyorsítani párhuzamos végrehajtással.

Az előző mérésben mennyi volt az exkluzív minták száma? Mennyi volt az inkluzív minták száma? Mit szimbolizál a $T_{\text{inkluzív}} - T_{\text{exkluzív}}$?

Adja meg zárt alakban a szűrő futásidejét (T_{filter}) a szűrő gyorsítását jelző *speedup* (mint változó) függvényében. A függvényben használja az előbbieken kimért exkluzív és inkluzív minták számát, mint konstansokat.

Grafikusan demonstrálja egy szakaszosan lineáris közelítéssel a szűrő futási idejét a *speedup* függvényében! Megéri a saját kódunkat a végtelenségig gyorsítani?

Javasolt maximális időráfordítás: maximum 15 perc.

A feladat érveléssel alátámasztva dokumentálandó.

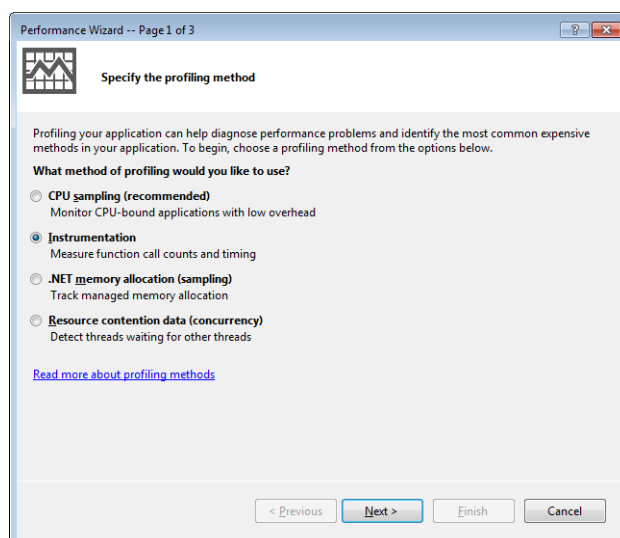
2.2 Instrumentálás

II/6. Feladat. Hajtson végre egy instrumentált profiling-ot a *steve1* mappában található képen, az Oil Painting, a Jitter és a Sobel edge detector szűrőket alkalmazva! Tekintse át a teljesítménymérés eredményének nézeteit! A következő feladatok az így előálló teljesítményjelentésre fognak vonatkozni.

Javasolt maximális időráfordítás: pár perc.

A feladatot nem szükséges dokumentálni.

(Ügyeljen rá, hogy a „kis függvények” kihagyása az instrumentációból csökkentheti a mérés eredményeinek felbontását; ezt egy már definiált mérési munkamenet kontextusmenüjében lehet szabályozni.)



6. ábra Visual Studio instrumentálás beállítások

II/7. Feladat.

Milyen alapvető különbségek figyelhetők meg a mintavételezett és az instrumentált nézetek között? Ugyanaz a mért adatok mértékegysége?

Miért tűnt el a „Show All Code” opció?

Javasolt maximális időráfordítás: 5 perc.

A feladat dokumentálandó.

II/8. Feladat.

Esetünkben mit fejez ki a System.Windows.Forms.Application.Run függvény exkluzív „eltelt ideje”?

Melyik korábban azonosított alkalmazás-állapotnak felel meg? (Ne feledjük: instrumentáció esetén pontos időmérésekkel és nem CPU-használat mintavételekkel rendelkezünk.)

Javasolt maximális időráfordítás: pár perc.

A feladat dokumentálandó.

II/9. Feladat. A steve2 mappában ugyanazon képet helyeztük el különböző felbontásokban. Már előre végrehajtottunk egy mérési kampányt, aminek adatai az asztalon a II-9 mappában találhatóak. A kampány során egy komplexebb szűrőt futtattunk.

Ábrázolja grafikusán a szűrő mért futásidejét egy változtatott bemeneti paraméter (pl. képszélesség/magasság vagy kép tárhelyigénye) függvényében! Az ábrázoláshoz mely bemeneti paramétert választotta?

Javasoljon függvény-típust mellyel a kettő közötti összefüggés jól leírható!

Javasolt maximális időráfordítás: 20-25 perc.

A feladat érveléssel alátámasztva dokumentálandó.

Ügyeljen arra, hogy különböző futásidejű, emberi interakciót is igénylő grafikus programok esetén a százalék dimenziójú futásidő-metrikák közvetlen összehasonlítása hibaforrásokat rejt.

Megjegyzés: az adatábrázolás eszközt szabadon megválaszthatják, de ügyeljenek a leadott képek áttekinthetőségére, értelmezhetőségére és minőségére.

II/10. Bónusz feladat. Próbálja meg ugyanezeket a méréseket önállóan is elvégezni. A kapott eredményeket hasonlítsa össze az általunk adottakkal. Hasonlóan skálázódik a futási idő?

Várható időráfordítás: 30+ perc.

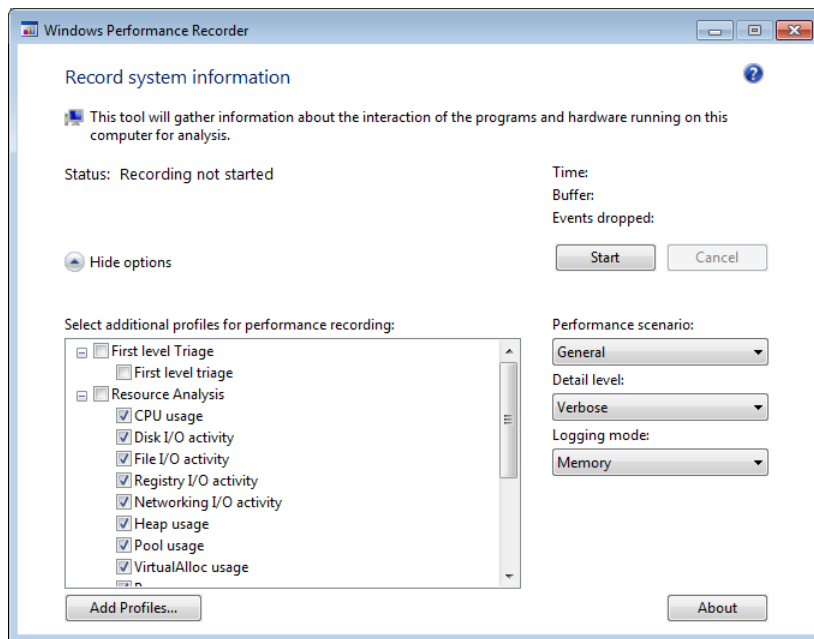
A feladat érveléssel alátámasztva dokumentálandó.

3 Nyomkövetés

III/1. Feladat. Indítson nyomkövetést a WPR segítségével, minden beépített „Resource Analysis” profilt bekapcsolva! A nyomkövetés során futtassa a FiltersDemo alkalmazást! A tárolt nyomot rögzítse állományba!

Javasolt maximális időráfordítás: pár perc.

A feladatot nem szükséges dokumentálni.



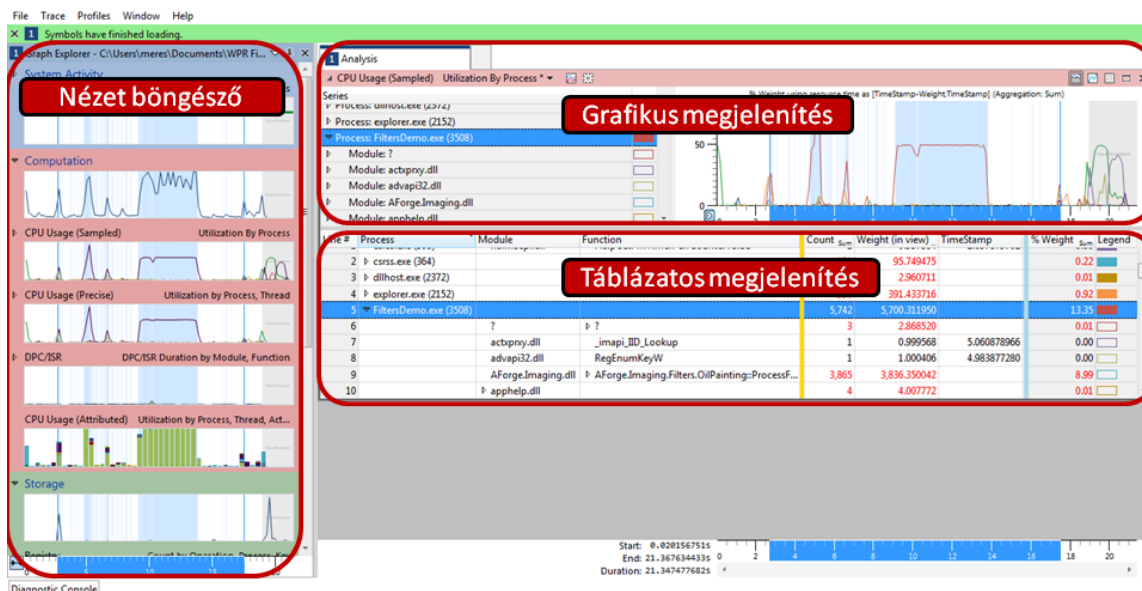
7. ábra Windows Performance Recorder beállítások

Figyelem: a memória-bufferek a “Cancel” gomb megnyomására rögtön törlődnek, így a mintákat még a nyomkövetés befejezése előtt mentjük. Javasolt továbbá egyetlen állományt betölteni és egyetlen szűrőt alkalmazni, majd kilépni az alkalmazásból. Ügyeljünk arra is, hogy lehetőleg ne generáljunk többletterhelést a platformon az alkalmazás mellé.

III/2. Feladat. Ismerkedjen meg a Windows Performance Analyzer-rel: indítsa el az eszközt és töltsse be az előző lépésben rögzített nyomot! Tekintse át a felület képességeit!

Javasolt maximális időráfordítás: 15 perc.

A feladatot nem szükséges dokumentálni.



8. ábra WPA nézetei

Javasolt munkamódszer (8. ábra):

- A nézet-böngésző szintjeinek kibontása és analízis lehetőségek áttekintése
- Nézetek egymás után analízis-fülre (jobb oldali sávra, „Analysis”) kihúzása és sorban megvizsgálása
 - o Grafikus megjelenítés beállítása és értelmezése
 - o Táblázatos megjelenítés beállítása és értelmezése (segítséget nyújthat egy későbbi feladat ábrája, a 10. ábra)

Megjegyzések:

- Táblázatos megjelenítés esetén (9. ábra) a vastag sárga vonaltól balra elhelyezkedő adatokat fa struktúrában lehet kibontani, ami a sárga vonaltól jobbra eső adatok csoportosítását határozza meg. A sárga és kék vonal közötti attribútumok nem jelennek meg a grafikonon, míg a kék vonaltól jobbra elhelyezkedő elemek igen. Azt, hogy mely attribútumok (oszlopok) jelenjenek meg, a „View Editorban” lehet konfigurálni (lásd fogaskerék ikon 10. ábra). A táblázatban elfoglalt pozíciójuk „fogd-és-vidd” módszerrel változtatható, a színes elválasztó vonalaktól függetlenül.
- A nézeteknek hasznos tulajdonsága, hogy ha egy objektumot kijelölünk („highlighting”), akkor az összes többi nézetben is a hozzátartozó adatok jelölődnek ki.
- A veremképet is rögzítő események esetében szükség lehet a rögzített szimbólumok feloldására (például függvényhívások neveinek megjelenítésekor). Ez a menüben a „Trace | Load Symbols” menüpont alatt kapcsolható be. Ez a szimbólumok betöltésére egyrészt helyi (PDB) fájlokat használ fel (amelyet a fordító is elhelyez egy lefordított .NET alkalmazás „Debug” könyvtárába), másrészt távoli szimbólumkiszolgálóktól gyűjtheti össze a szükséges adatokat. Utóbbi eljárás segítségével a Windows binárisok hívásainak többsége is követhető lesz. Jó néhány olyan binárist futtat azonban a virtuális gép, melyekhez nem érhetőek el publikusan szimbólumadatok.
- Senkit ne riasszon el, ha az attribútumok és fogalmak egy jelentékeny része ismeretlennek bizonyul. Az ETW infrastruktúráról tudni kell, hogy azt a Microsoft eredetileg a Windows, mint operációs rendszer belső, operációsrendszer-fejlesztést támogató nyomkövetéséhez kezdte el kifejleszteni. Így az általa nyerhető betekintés mélysége és „szélessége” igen nagy; az összes adat értelmezéséhez a Windows kernel olyan szintű ismerete szükséges, melyet nyilvánvalóan nem követelünk meg.

Line #	Process	Module	Function	Count _{sum}	Weight (in view)	TimeStamp	% Weight _{sum}	Legend
5	FiltersDemo.exe (35...)			5,742	5,700.311950		13.35	
6		?	?	3	2,868520		0.01	
7		actxprny.dll	_imapi_IID_Lookup	1	0.999568	5.060878960	0.00	
8		advapi32.dll	RegEnumKeyW	1	1.000406	4.983877280	0.00	
9		AForge.Imaging.dll	AForge.Imaging.Filters.OilPainting::ProcessF...	3,865	3,836.350042		8.99	
10		apphelp.dll		4	4.007772		0.01	
11		clbcatq.dll	IsComplusComponent	1	0.997334	5.018887870	0.00	
12		clr.dll		752	742.510163		1.74	
13			XML Stream::parseTable	1	1.000407	2.684878400	0.00	

9. ábra Táblázatos nézet

III/3. Feladat. Viselkedés felderítése: CPU.

Konfiguráljon fel CPU Usage (Sampled) nézetet, mely az alkalmazás CPU-használatát mutatja grafikusán.

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.

Grafikusan azonosítsa a filter-hívást a teljes CPU-használatban kijelölések („highlighting”) segítségével. Képernyőképpel támassza alá, amin a táblázatos nézetben látszik a kijelölt függvény (amihez feladat a szimbólumok feloldása, lásd a megjegyzést fentebb), illetve ennek hatására a grafikonon a CPU-használat.

Javasolt maximális időráfordítás: 10 perc.

A feladat dokumentálandó. Képernyőkép és rövid indoklás is szükséges.

Megjegyzés: ahogyan a nézeteket is az analízis fülre lehetett húzni, úgy a táblázat oszlopait is „drag-n-drop” át lehet sorrendezni.

View Editor (nézet szerkesztő) megnyitása (oszlopok konfigurálása)

Táblázat és/vagy grafikon nézet beállítása

Modulok és függvények mutatása a nézetben

Available Columns	Visible	Name	Aggregation	Sort
% Weight	<input type="checkbox"/>	Process Name	None	None
Address	<input type="checkbox"/>	Display Name	None	None
All Count	<input checked="" type="checkbox"/>	Process	None	None
Annotation	<input type="checkbox"/>	Stack Tag		
Compiler Optimization	<input type="checkbox"/>	Stack (Frame Tags)		
Count	<input type="checkbox"/>	Stack		
DPC/ISR	<input checked="" type="checkbox"/>	Module	None	None
Function	<input checked="" type="checkbox"/>	Function	None	None
Inlined Functions	<input type="checkbox"/>	DPC/ISR	None	None
Is PGO'ed	<input type="checkbox"/>	Address	None	None
Module	<input type="checkbox"/>	Thread ID	None	None
PGO Counts	<input type="checkbox"/>	Thread Name	None	None
PGO Dynamic Instruction Co	<input type="checkbox"/>	Thread Activity Tag	None	None
Priority	<input type="checkbox"/>			
Process	<input type="checkbox"/>			

10. ábra CPU nézet beállítása

III/4. Feladat. Diszk és fájl I/O használat elemzése előre elkészített trace alapján.

Nyissa meg az asztalon a III-5 mappában található nyomot (trace). Konfiguráljon fel egy nézetet, mely 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. A oszlopokat értelemszerűen rendezze.)

Mely jpg kiterjesztésű fájlokat olvastuk be a FiltersDemo programmal? Vizsgálja meg a fájl műveleteket (File I/O nézet)!

Mely jpg kiterjesztésű fájl(ok)hoz nyúlt az operációs rendszer a merevlemezhez?

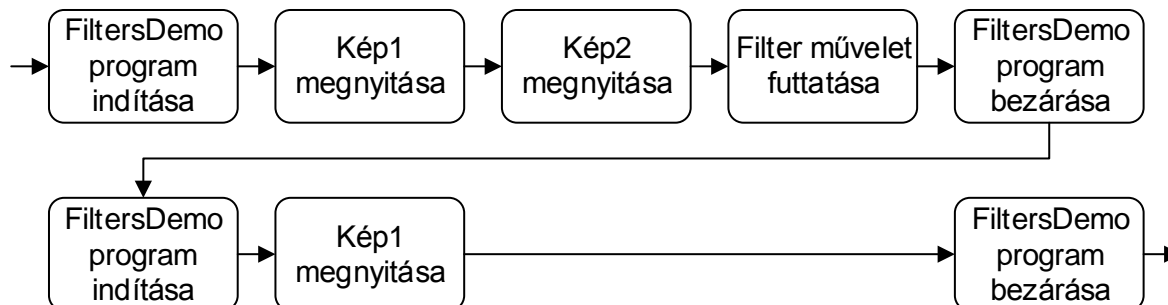
Szűrjön (Filter to Selection) mindkét nézetben arra a fájlra, amikor merevlemez művelet is volt. (Fontos: ne az Event Type, hanem a File Name alapján szűrjön!) Ugyanazt mutatja a két grafikon? Mivel magyarázható a jelenség?

A folyamatábráról (11. ábra) is látszik, hogy a „Kép1”-ként hivatkozott fájlt kétszer nyitottuk meg. A fájl I/O időket vizsgálva mit tapasztalunk? Mivel magyarázható a jelenség?

Javasolt maximális időráfordítás: 25 perc.

A feladat dokumentálandó. Képernyőkép és rövid indoklás is szükséges.

Megjegyzés: a nézetek konfigurációja (pl. melyik attribútumok szerepelnek és miért, melyek nem szerepelnek és miért nem) egyfajta tervezői (még inkább elemzői) döntést jelentenek, így a szükséges (szűk) terjedelemben dokumentálandóak.



11. ábra FilterDemo futtatási folyamatábrája