

## Практична робота №2.

### Дослідження режимів роботи обчислювальної системи та обробки переривань в ОС Windows за допомогою консолі Performance Monitor

**Мета роботи** — одержання практичних навичок використання консолі Performance Monitor.

- Робота виконується на основній або віртуальній машині із ОС Windows.
- Використовуються системні утиліти: консоль Монітор продуктивності (Performance Monitor), Диспетчер завдань (Task Manager).

### Теоретичні відомості

#### Режими роботи обчислювальної системи

Для запобігання доступу додатків до критично важливих даних операційної системи та усунення ризику їх модифікації Windows використовує два режими доступу до процесора (навіть, якщо він підтримує більше двох режимів): *користувацький* (user mode) і *ядра* (kernel mode). Код додатків працює в режимі користувача, тоді як код операційної системи (наприклад, системні сервіси і драйвери пристроїв) — у режимі ядра. У режимі ядра надається доступ до всієї системної пам'яті і дозволяється виконувати будь-які машинні команди процесора.

Хоча кожен Windows-процес має свою (закриту) пам'ять, код операційної системи і драйвери пристроїв, що працюють в режимі ядра, ділять єдиний віртуальний адресний простір. Кожна сторінка віртуальної пам'яті позначається тегом, який визначає, у якому режимі повинен працювати процесор для читання і/або запису даної сторінки. Сторінки в системному просторі доступні лише у режимі ядра, а всі сторінки в користувацькому адресному просторі — у режимі, який призначений для користувача.

Windows не передбачає ніякого захисту системної пам'яті від компонентів, що працюють у режимі ядра. Інакше кажучи, код операційної системи і драйверів пристроїв в режимі ядра отримує повний доступ до системної пам'яті і може обходити засоби захисту Windows для звернення до будь-яких об'єктів. Оскільки основна

частина коду Windows виконується в режимі ядра, вкрай важливо, щоб компоненти, що працюють у цьому режимі, були ретельно продумані і протестовані.

Це також підкреслює, наскільки необхідно бути обережним при завантаженні драйвера пристрою від стороннього постачальника: перейшовши в режим ядра, він отримує повний доступ до всіх даних операційної системи. Така вразливість стала однією з причин, за якими в Windows введено механізм перевірки цифрових підписів драйверів, що попереджає користувача про спробу установки неавторизованого (непідписаного) драйвера.

Для взаємодії користувача та обчислювальної системи Windows надає зручний графічний інтерфейс. Віконна і графічна підсистема, що відповідає за функціонування інтерфейсу у Windows, працює в режимі ядра. Тому додатки, які інтенсивно працюють із графікою, більшу частину часу перебувають саме в режимі ядра, а не у режимі користувача.

Ця підсистема складається із декількох рівнів: менеджер вікон, інтерфейс графічних пристроїв, драйвери графічних пристроїв.

Менеджер вікон реалізує високорівневі функції. Він керує віконним виведенням, обробляє введення з клавіатури або миші та передає додаткам повідомлення користувача.

Інтерфейс графічних пристроїв (Graphical Device Interface, GDI) складається із набору базових операцій графічного виведення, які не залежать від конкретного пристрою (креслення ліній, відображення тексту тощо).

Драйвери графічних пристроїв (відеокарт, принтерів тощо) відповідають за взаємодію з контролерами цих пристроїв.

Під час створення вікон або елементів керування запит надходить до менеджера вікон, який для виконання базових графічних операцій звертається до GDI. Потім запит передається драйверу пристрою, далі — апаратному забезпеченню через HAL.

### Апаратні переривання, системні виклики, виключні ситуації

Система переривань є основним механізмом, що забезпечує функціонування ОС. За допомогою переривань процесор отримує інформацію про події, не пов'язані з основним циклом його роботи (отриманням інструкцій з пам'яті та їх виконанням). Переривання бувають двох типів: апаратні і програмні.

Апаратне переривання — це спеціальний сигнал (irq, interrupt request, запит переривання), що передається процесору від апаратного пристрою.

До апаратних переривань належать:

- переривання введення-виведення, що надходять від контролера периферійного пристрою; наприклад, таке переривання генерує контролер клавіатури при натисканні на клавішу;
- переривання, пов'язані з апаратними помилками (такі переривання виникають, наприклад, у разі збою контролера диска).

Системний виклик — це засіб доступу до певної функції ядра ОС із прикладних програм. Набір системних викликів визначає дії, які ядро може виконати за запитом процесів користувача. Системні виклики задають інтерфейс між прикладною програмою і ядром ОС.

Виключна (виняткова) ситуація — подія, що виникає в результаті спроби виконання програмою команди, яка з якихось причин не може бути виконана до кінця. Прикладами таких команд можуть бути спроби доступу до ресурсу за відсутності достатніх привілеїв чи звернення до відсутньої сторінки пам'яті, ділення на нуль.

### Performance Monitor

Монітор продуктивності (Performance Monitor) – консоль Windows, що використовує лічильники, кожен з яких пов'язаний із (і стежить за) певними об'єктами в операційній системі. Приклад лічильників, з якими працює Монітор ресурсів: кількість процесів, які очікують на роботу з дисками, відсоток завантаженості процесора, кількість пакетів, що проходять через мережу за секунду. Консоль може збирати дані у файл, малювати діаграми або надсилати повідомлення в разі досягнення лічильником певного значення. У можливості монітора входить:

- перегляд даних одночасно від будь-якої кількості комп'ютерів;
- негайна видача відгуку після внесення змін в систему.
- оперативний аналіз – перегляд та динамічна зміна графіків, що відображають показники поточної активності системи (Продуктивність – Засоби спостереження – Системний монітор);

- експорт даних із графіків, log-файлів і звітів у бази даних і електронні таблиці (наприклад: Excel, Access, SQL Server);
- створення порогових значень для будь-яких лічильників, у разі досягнення яких з'явиться запис в log-файлі і за бажанням буде відправлено повідомлення користувачу;
- створення log-файлів для даних із різних комп'ютерів, що включають різні (або однакові) лічильники, протягом різних (або однакових) проміжків часу для подальшого спільного аналізу;
- дописування до існуючих log-файлів для організації довготривалої статистики;
- збереження конфігурації, графіка або звіту для подальшого використання.

#### Спостереження за режимами роботи обчислювальної системи

Ситуація, коли користувальницький потік частину свого часу працює в режимі користувача, а частина — в режимі ядра, абсолютно нормальна. Оскільки підсистема, що відповідає за підтримку графіки і вікон в ОС Windows, функціонує в режимі ядра, то додатки, які інтенсивно працюють із графікою, більшу частину часу діють у режимі ядра, а не в режимі користувача. Найпростіший спосіб перевірити це — запустити додаток на зразок Microsoft Paint і за допомогою лічильників оснастки Performance, що наведені у таблиці 1, поспостерігати за показниками часу роботи процесора в режимі користувача і в режимі ядра.

Таблиця 1. Лічильники консолі Performance Monitor

№	Об'єкт: Лічильник	Опис
1	Processor:% Processor Time Процесор: % Завантаженості процесора	процент часу, який процесор витрачає на обробку всіх потоків команд, крім простою максимально допустимим значенням відсотку завантаженості процесору є <b>85%</b> — тривалі перевищення цього значення є приводом для встановлення більш швидкого процесору або для установки додаткових процесорів

2	Processor: % Idle Time Процесор: % Часу простою	процент часу, який процесор простоює протягом інтервалу вибірки
---	---	---

Таблиця 1. Лічильники консолі Performance Monitor (продовження)

3	Processor: %Privileged Time Процесор: % Роботи в привілейованому режимі	процент часу, на протязі якого окремий процесор (чи всі процесори) працюють у режимі ядра
4	Processor: %User Time Процесор: % Роботи в режимі користувача	процент часу, на протязі якого окремий процесор (чи всі процесори) працюють у режимі користувача
5	Process\Idle:%Privileged Time Процес\Idle: %Роботи в привілейованому режимі	процент часу, на протязі якого процес Idle працює у режимі ядра
6	Process\Idle:% User Time% Процес\Idle: %Роботи в режимі користувача	процент часу, на протязі якого процес Idle працює у режимі користувача

### *Системний процес Idle*

Процесор завжди повинен виконувати код. Якщо процесор не виконує процес, що містить корисний код, то в ОС Windows він виконує процес Idle. Це забезпечує зайнятість процесора. Процес Idle має по одній нитці для виконання на кожному з процесорів і майже 100% свого часу перебуває в режимі ядра. Він використовується для обліку тактів процесора у стані простою. Бездіяльність системи використовується Windows для зниження енергоспоживання процесора. Конкретна схема зниженого енергоспоживання визначається апаратним забезпеченням і можливостями мікропрограми системи. Наприклад, на x86 процесорах цей процес буде виконувати

в циклі інструкцію HLT, яка вказує процесору відключити деякі внутрішні компоненти і чекати апаратного переривання.

### *Диспетчер завдань*

За тією ж активністю перемикаць процесору між режимом ядра та користувача можна поспостерігати через Task Manager (Диспетчер завдань). Просто перейдіть у ньому на вкладку Performance (Швидкодія), а потім виберіть із меню View (Вид) команду Show Kernel Times (Відобразити час ядра). Відсоток завантаженості процесора відображається зеленим кольором, а відсоток часу роботи в режимі ядра — червоним.

### Спостереження за обробкою переривань

Для спостереження за частотою переривань використовуються лічильники, наведені в табл.2.

Таблиця 2. Лічильники консолі Performance Monitor

№	Об'єкт: Лічильник	Опис
1	Processor: Interrupts /sec Процесор: Переривань/сек	частота, з якою процесор отримує і обслуговує апаратні переривання  максимально допустиме значення кількості переривань залежить від покоління процесору
2	Processor: %Interrupt Time Процесор (%Переривань)	відсоток часу, який процесор витрачає на обробку апаратних переривань  максимально допустимим значенням є <b>20-30%</b> — значне і тривале перевищення цього порогу без адекватного завантаження пристроїв вказує на апаратні проблеми

Таблиця 2. Лічильники консолі Performance Monitor (продовження)

3	System: System Calls/sec Система: Системних викликів/сек	частота звернень до процедур системних служб операційної системи з боку всіх процесів, які виконуються на комп'ютері
4	System: Exceptions/ Sec Система:Обробка виключних ситуацій/сек	частота виконуваної системою обробки виключних ситуацій (exceptions)

### Завдання на практичну роботу

1. Побудова лічильників продуктивності із таблиці 1 для дослідження режимів роботи процесора.

2. Побудова лічильників продуктивності із таблиці 2 для дослідження обробки переривань.

3. Збір даних при відсутності взаємодії із системою (інтервал оновлення — 5 с, інтервал спостереження – 1 хв.). Після виконання цього завдання повинно бути 2 відформатованих рисунки gif та 2 файли .blg (перші рисунок та файл відповідають відсотковим значенням, другі — чисельним).

4. Збір даних при взаємодії із обраним процесом (інтервал оновлення — 5 с, інтервал спостереження – 1 хв.). Після виконання цього завдання повинно бути 2 відформатованих рисунки gif та 2 файли .blg (перші рисунок та файл відповідають відсотковим значенням, другі — чисельним).

5. Створення електронного звіту.

5.1. *Дослідження відсотку роботи процесору у режимах ядра та користувача.*

5.2.1. На одній сторінці розмістіть 2 рисунки із графіками лічильників таблиці 1, отриманих в п. 3 та 4.

5.2.2. На основі рисунків потрібно заповнити відповідними даними таблиці 3 та 4 (середнє та максимальне значення лічильника будується автоматично).

5.2.3. Зробіть висновок, у якому режимі (користувача чи ядра) здебільшого працював процесор, чи залежить це від процесів, які на ньому виконувалися? Які дії досліджуваній вами процес виконував у режимі користувача, а що переадресовував на виконання ОС? Чи не перевищує показник Processor:%Processor Time рекомендованого максимального значення?

5.2. Дослідження роботи операційної системи по обробці переривань

5.2.1. На одній сторінці розмістіть 2 рисунки із графіками лічильників таблиці 2, отриманих в п. 3 та 4.

5.2.2. На основі рисунків потрібно заповнити відповідними даними таблиці 3 та 4.

5.2.3. Зробіть висновки, чи не перевищує показник Processor:%Interrupt Time рекомендованого максимального значення.

Таблиця 3. Середнє значення лічильників

Лічильник	Середнє значення лічильника	
	відсутність взаємодії, %	взаємодія з обраним процесом,%
Processor:% Processor Time		
Processor: % Idle Time		
Processor: %Privileged Time		
Processor: %User Time		
Process\Idle: %Privileged Time		
Process\Idle:% User Time		
Processor: Interrupts /sec		
Processor: %Interrupt Time		
System: System Calls/sec		
System: Exceptions/ Sec		



Таблиця 4. Максимальне значення лічильників

Лічильник	Максимальне значення лічильника		Тривалість перевищення, с
	відсутність взаємодії, %	взаємодія з процесом, %	
Processor:% Processor Time			
Processor: %Interrupt Time			

Таблиця 5. Перелік процесів для взаємодії

Остання цифра номеру студента за списком визначає номер його варіанта

№ варіанта	Програма
1.	Текстовий редактор
2.	Файловий менеджер
3.	Браузер
4.	Гра
5.	Калькулятор
6.	Графічний редактор
7.	Середовище розробки
8.	VirtulBox
9.	Програма для пошти
0	Торент-клієнт

### Підготовка до виконання практичної роботи

Ознайомтеся з рекомендаціями до виконання практичної роботи №2, які розміщено у методичних вказівках «Операційна система Windows. Частина 2», що доступні за адресою <http://fpm.kpi.ua/archive/dir.do>.

### Вимоги до оформлення результатів роботи

Електронний звіт про практичну роботу повинен мати назву N.doc, де N – номер студента за списком, і містити:

- 1) титульний аркуш;
- 2) 4 рисунки;

- 3) заповнені таблиці 3, 4;
- 4) висновки з виконання практичної роботи.

До захисту друкуються пункти 1, 3 електронного звіту.

Електронний звіт викладається на Системі Moodle електронного порталу коледжу.

### **Питання для самоперевірки**

1. Характеристики режиму ядра.
2. Для чого використовується процес Idle в ОС Windows?
3. Як можна прослідкувати за часом роботи у режимі користувача та ядра за допомогою Task Manager?
4. Які можливості утиліти Performance Monitor з оперативного спостереження за станом обчислювальної системи?
5. Які можливості утиліти Performance Monitor зі створення звітів?
6. Які граничні значення лічильників вказують на можливі негаразди в роботі системи?

### **Рекомендована література**

1. Шеховцов В.А. Операційні системи. — К .: Видавнича група ВНУ, 2005. — 576 с.
2. <https://technet.microsoft.com/ru-ru/magazine/2008.08.pulse.aspx>