

## РОЗРОБКА АЛГОРИТМУ ВІДСТЕЖЕННЯ АКТИВНИХ ПРОЦЕСІВ У ОПЕРАЦІЙНІЙ СИСТЕМІ WINDOWS

<sup>1</sup>Вінницький національний технічний університет

### Анотація

*Розглянуто алгоритм відстеження активних процесів у операційній системі Windows. Проаналізовано способи вдосконалення алгоритму та його реалізації.*

**Ключові слова:** алгоритм, відстеження, функція, продуктивність, процес, засіб, віддалена робота.

### Abstract

*This article covers an overview of algorithm for tracking process in Windows operating system. There was conducted analysis on possible ways for enhancement and implementation of the algorithm.*

**Keywords:** algorithm, tracking, function, productivity, process, utility, remote work.

### Вступ

Внаслідок пандемії вірусу COVID-19 у світі став поширений режим роботи з дому. Пандемія відступила, але багато працівників досі надають перевагу роботі з дому чи гібридному режиму відвідування офісу. На додачу, внаслідок проведеного опитування аналітичною організацією Gartner серед 127 керівників компаній, було досліджено, що 82% лідерів компаній дозволятимуть підлеглим працювати з дому час від часу, а 47% на постійній основі [1].

Проте внаслідок переходу на віддалену роботу робочий день працівника схильний затримуватися. Згідно дослідження організації National Bureau of Economic Research середня тривалість робочого дня подовжилась на 48,5 хвилин протягом тижнів локдауну, а кількість зустрічей зросла на 13 відсотків [2].

По суті у тижневі з'являється майже повноцінний робочий день. Як наслідок, це може призвести до професійного вигорання працівників та до погіршення якості виконаної роботи. Одним з рішень, що можуть допомогти запобігти цьому, є відстеження роботи працівників за комп'ютером. Таким чином, відслідковуючи робочий час користувачів, можна запобігти марній витраті часу, простроченню дедлайнів, покращити концентрацію на робочих задачах та загалом краще приймати організаційні рішення з огляду на відстежені дані роботи.

Зокрема, відстеження роботи користувача є базовою вимогою до застосунків із контролю та підвищення продуктивності користувача [3]. Тому актуальною задачею є розробка алгоритму відстеження активних процесів у операційній системі для програми контролю роботи користувача.

Метою роботи є оптимізація та вдосконалення алгоритму відстеження роботи користувача за комп'ютером, виконавши аналіз відповідних методів релаксації.

Об'єктом дослідження є процес відстеження активних процесів у операційній системі.

Предметом дослідження є алгоритми та засоби реалізації відстеження активних програм в операційній системі.

### Основна частина

Багато операційних систем мають вбудовані засоби для відстеження процесів у системі. Зокрема, в операційній Windows є вбудовані такі утиліти:

- Event Viewer - дозволяє переглядати системні логи та події;
- Task Manager - надає перелік активних процесів, використання ними ресурсів і продуктивності системи;
- TaskList – вбудована утиліта операційної системи Windows, що дозволяє отримувати інформацію про різні властивості активних процесів, такі як назви процесів, ідентифікатори процесів (PID), використання пам'яті, навантаженість процесора тощо.

Кожна з цих утиліт має свої недоліки у реалізації алгоритму відстеження роботи користувача. Так, у EventViewer чи Task Manager немає API, яке дозволить програмним шляхом отримати властивості процесів у системі.

Утиліта TaskList може програмно використовуватися для отримання даних про активні процеси у системі, проте на кожне застосування запускається командний рядок Windows, що ускладнює роботу програми та може спантеличити користувача.

З огляду на недоліки вищезгаданих засобів відстеження процесів необхідною є програмна реалізація відповідного алгоритму за допомогою бібліотек системних функцій керування. Наприклад, у мові програмування Python є бібліотека psutil, що містить засоби для отримання інформації про активні процеси, хоча й потребує попереднього завантаження. На додачу, мова програмування C++ теж містить вбудовану бібліотеку psapi.h та windows.h для керування процесами операційної системи Windows [4]. Оскільки C++ широко використовується для системного програмування, надалі нею розглядатиметься реалізація алгоритму відстеження активних процесів.

Загалом для того, щоб відстежити активні процеси, необхідно виконати такі кроки:

1. Ініціалізувати структуру даних, у якій планується збереження імен процесів.
2. Отримання ідентифікаторів процесів та сумарного розміру пам'яті, що вони займають за допомогою функції EnumProcesses бібліотеки psapi.h.
3. Обрахування числа активних процесів за допомогою виконання операції ділення сумарного розміру пам'яті, що займають процеси на розмір типу даних DWORD.
4. Перебрати ідентифікатори процесів у циклі та ініціалізувати дескриптори активних процесів за допомогою функції OpenProcess.
5. За допомогою методу GetModuleBaseName отримати назву процесу та перевірити чи він уже присутній у списку поточних процесів. Якщо ні, то процес повинен додатися до списку активних процесів.
6. Закрити дескриптор процесу для розвантаження ресурсів операційної системи.

Внаслідок послідовного виконання цих кроків можна програмно отримати список поточних процесів у системі. Щоб постійно відстежувати активні процеси у потоці необхідно реалізовувати вищеписаний алгоритм в окремому потоці виконання. Це дозволить користувачу працювати із програмою та безперебійно відстежувати активні програми.

В окремому потоці даних необхідно також створити список усіх програм, що запускались впродовж користувацької сесії для того, щоб підсумовувати час роботи програм в умовах динамічних оновлень.

## Висновки

Таким чином, було розглянуто принцип роботи алгоритму відстеження активних процесів у системі та його реалізацію, використовуючи вбудовані засоби операційної системи Windows, мови програмування Python та C++. Було обґрунтовано важливість розробки такого алгоритму та покроково описано способи реалізації такого алгоритму. Застосування запропонованого алгоритму в програмі для відстеження роботи користувача дозволить реалізувати базову вимогу до застосунку для підвищення продуктивності користувача, а саме відстеження роботи користувача.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Gartner Survey Reveals 82% of Company Leaders Plan to Allow Employees to Work Remotely Some of the Time [Електронний ресурс] // Gartner – 2020. – Режим доступу до ресурсу: <https://www.gartner.com/en/newsroom/press-releases/2020-07-14-gartner-survey-reveals-82-percent-of-company-leaders-plan-to-allow-employees-to-work-remotely-some-of-the-time>
2. Collaborating During Coronavirus: The Impact of COVID-19 on the Nature of Work [Електронний ресурс] // NBER | National Bureau of Economic Research. – 2020. – Режим доступу до ресурсу: <https://www.nber.org/papers/w27612>
3. Дем'яник І.В. Вимоги до сучасних програмних застосунків для тайм-менеджменту/ І.В. Дем'яник, О. В. Романюк // Матеріали конференції «ЛІІ Науково-технічна конференція підрозділів Вінницького національного технічного університету (2023)», Вінниця, 2023. Режим доступу до ресурсу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2023/paper/view/17495/14615>.
4. Psapi.h header [Електронний ресурс] // Microsoft – 2023. – Режим доступу до ресурсу: <https://learn.microsoft.com/en-us/windows/win32/api/psapi/>

*Дем'яник Ірина Валеріївна* – студентка групи ІПІ-19б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: [ira.demianyk10@gmail.com](mailto:ira.demianyk10@gmail.com).

*Романюк Оксана Володимирівна* – к.т.н., доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: [romaniukoksanav@gmail.com](mailto:romaniukoksanav@gmail.com).

*Iryna Demianyk* – student of group ІPI-19b, Faculty for Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: [ira.demianyk10@gmail.com](mailto:ira.demianyk10@gmail.com).

*Oksana Romaniuk* – Candidate of Technical Sciences, Associate Professor of the Software Chair, Vinnytsia National Technical University, Vinnytsia, e-mail: [romaniukoksanav@gmail.com](mailto:romaniukoksanav@gmail.com)