

Вказані засади мають стати основою для побудови суб'єктами господарювання комплексу конкретних заходів щодо прогнозування, ідентифікації та протидії загрозам недобросовісних дій з боку конкурентів на основі наявних технічних можливостей, людських та інформаційних ресурсів, а також нормативно-правового базису.

### ***Список використаних джерел:***

1. Про захист від недобросовісної конкуренції: Закон України № 236/96-ВР в редакції від 11.12.2019, підстава - 286-ІХ. URL: <https://zakon.rada.gov.ua/laws/show/236/96-%D0%B2%D1%80#Text>

2. Харченко О., Яремич В. Протидія кібератакам, базованим на прийомах соціальної інженерії URL: <http://surl.li/arnjt>

**Ключові слова:** конкуренція, кібербезпека, особистий бренд, персональні дані.

**Ключевые слова:** конкуренция, кибербезопасность, личный бренд, персональные данные.

**Key words:** competition, cybersecurity, personal brand, personal data.

### ***Трофименко Олена Григорівна***

*Національний університет «Одеська юридична академія»,  
доцент кафедри інформаційних технологій, кандидат технічних наук,  
доцент*

### ***Чепендюк Олександр Олександрович***

*Національний університет «Одеська юридична академія»,  
студент 2-го курсу факультету кібербезпеки та інформаційних технологій*

## **ЧОМУ C++?**

Хоча мова програмування C++ з'явилася в далеких 80-х [1], ще й досі вона актуальна, завдяки своїм унікальним перевагам. Не зважаючи на поширені і

тривалі запевнення про те, що С++ ось-ось помре, ця мова і нині входить в топ-10 [2], а то і в топ-5 [3] рейтингів найзатребуваніших мов програмування на ринку ІТ-праці.

Ця мова дає можливість розробнику максимально контролювати всі аспекти структур, підтримує взаємодію з пам'яттю на апаратному рівні і саме тому підходить для створення проєктів, в яких важлива швидкість і продуктивність. Поширено С++ застосовується для розробки системних драйверів, сучасних операційних систем, середовищ розробки, веб-браузерів, графічних редакторів тощо. Наприклад, операційні системи Windows та macOS мають елементи, написані засобами С++, а YouTube використовує С++ для обробки відео [4]. Крім того, С++ використовують при створенні реальної приватної та публічної інфраструктури блокчейну, для розробки шейдерів для графічного програмного забезпечення (ПЗ) та ігор, а також для вивчення нейромереж. С++ потрібна для розробок вбудованих (embedded) систем, щоб пов'язати «залізо» з програмним забезпеченням. Спектр застосування С++ вражає, адже програмовані вбудовані системи використовуються повсюдно: від пристроїв на нашій кухні до промисловості, сільського господарства, медицини, фінансової сфери, автомобілебудування, космічної галузі та фундаментальних досліджень у науці [4, 5]. Розумні будинки, системи моніторингу (від відеонянь до операційних), розробка серверів для аудіо/відеоконференцій, медіаплеєри потребують програмних рішень, для яких мова С++ є основною. Тому С++-розробники будуть ще довго затребувані і створювати справді круті проєкти, що змінюватимуть наш світ на краще та якісно покращуватимуть життя людей.

Мова С++ швидка й універсальна. Її творець Бйорн Страуструп розробив цю мову як статично типізовану, мультипарадигмальну універсальну мову програмування середнього рівня складності на основі мови С [5]. Оскільки це розширена версія мови програмування С, С та С++ часто позначають разом як С/С++.

Для кращого розуміння живучості С++ розберемося з її перевагами:

– *швидкість* – одна з головних переваг С++. Швидшою за С++ із поширених зараз мов програмування є лише мова С. Як швидкодієві засоби

програмної розробки для реалізації локальних потреб можуть підійти мови C, Go або Rust, але за допомогою C++ можна писати програми будь-якої складності з гарною швидкодією. Важливо, що йдеться не про час написання коду, який, наприклад, у мов Python, C# і Java може бути суттєво меншим, а код компактнішим. Проте кінцевого користувача мало цікавить час розробки програми, на відміну від зацікавленості у швидкісній роботі відповідного програмного застосунку;

- *кросплатформність* – можливість розробляти програми для різних платформ та операційних систем. C++ є достатньо універсальною, адже її компілятори є в кожній операційній системі, і більшу частину програм можна перенести з однієї платформи на іншу без проблем із бібліотеками та середовищем розробки. При цьому нині важко знайти програмний застосунок, повністю написаний лише засобами C++, але частково його використовує практично кожен більш менш серйозний програмний проєкт. Тому знання цієї мови, чи хоча б певних навичок роботи з нею, ніколи не будуть зайвими для розробника ПЗ;

- підтримка численних парадигм програмування: процедурне програмування, модульне програмування, абстракція даних, об'єктно-орієнтоване програмування та узагальнене програмування. C++ допомагає зрозуміти різні підходи до програмування та методології розробки, які можна застосовувати і в інших мовах. Саме через це більшість навчальних закладів вищої освіти, які готують IT-фахівців, мають курс по вивченню C++;

- можливість роботи на апаратному рівні з портами, доступ до API операційної системи, безпосередній доступ до пам'яті комп'ютера, з одного боку, є потужними засобами розробки, а з іншого – ці могутні механізми вимагають певної культури програмування, адже їхнє безконтрольне використання небезпечно в невмілих руках і може спричинити помилки. Проте не варто називати молоток небезпечним інструментом, коли хтось не вміє користуватися ним. Довільна арифметика вказівників, механізми виділення і вивільнення пам'яті, конструктори і деструктори, стандартні шаблони, передача параметрів за посиланням дозволяють продуктивно працювати з пам'яттю на апаратному

рівні. А відсутність в C++ засобів автоматичної перевірки вказівників на вихід за межі діапазону вимагає значної відповідальності і кваліфікації розробника.

C++ доречний для проектів, для яких важливою є продуктивність коду. Ця мова не ідеальна, адже не існує ідеальних мов програмування. Наявність безлічі можливостей, що порушують принципи типобезпеки, може спричинити важковловимі помилки в C++-програмах. Більшість проблем типобезпеки C++ успадкована від C. Відсутність у C++ вбудованої системи прибирання сміття, з одного боку, не рідко через неухважність, особливо у новачків, спричиняє помилки типу «переповнення буфера», а з іншого – відмова автора мови від ідеї використовувати автоматичне керування пам'яттю дозволяє користувачеві самому вибрати стратегію керування ресурсами і завдячує гарній швидкості виконання коду, що є певною візитівкою C++. Також слід зважати на те, що долучення через директиву препроцесора `#include` до інтерфейсу великої кількості зовнішніх модулів відчутно уповільнює компіляцію. Тому для усунення цього специфічного недоліку поширено реалізується механізм перекompіляції заголовних файлів (англ. *Precompiled Headers*).

Багато хто жаліється на складність вивчення мови C++. При цьому в основі популярних Java, JavaScript, C# закладено синтаксис мови C++. Тому специфіку роботи цих мов достатньо не просто опанувати, якщо розробник не знайомий з плюсами. Саме через це C++ тривалий час є однією з перших мов у навчанні програмуванню. Так, за даними дослідження GlobalLogic у Львові, 30% студентів визначили основним напрямком саме C++ [4]. Переважно причиною критики C++ є невміння роботи з нею, тобто небажання враховувати наслідки безконтрольного використання наявних свобод у конструкціях мови, технічну відсутність обмежень на використання її засобів. Адже ця мова вибаглива до знання та врахування її специфіки та тонкощів, але при цьому дозволяє створювати ті об'єкти, які не можливо створювати засобами інших мов програмування. До того ж C/C++ дозволяє проводити агресивну оптимізацію такого рівня, який є недосяжним для інших мов високого рівня.

Мова C++ жива і постійно розвивається, щоб відповідати сучасним вимогам. Офіційна стандартизація C++ розпочалася 1998 року, коли був вперше

опублікований стандарт мови ISO/IEC 14882:1998 (більш відомий як C++98). Наступний стандарт C++ ISO/IEC 14882:2003 був опублікований 2003 року, а переглянутий 2009 року. У 2011 з'явився стандарт C++11, у 2014 – C++14, у 2017 – C++17, а в 2020 – C++20, який, до речі, є діючою версією стандарту [6]. З кожною новою версією додаються нові можливості, конструкції мови ускладнюються, але сам код при цьому стає більш зручним для читання.

З усього вище зазначеного можна зробити висновок, що C++ як мова програмування і досі перспективна, не зважаючи на те, що існують більш нові високорівневі мови. Зараз професійним розробникам ПЗ варто знати кілька мов, адже не існує ідеальних засобів розробки, та й не повинна мова загального призначення бути ідеальною для вирішення всіх завдань. Те, що є найкращим засобом розробки для одного завдання, часто виявляється недоречним тягарем для іншого.

У C++ є свої переваги і недоліки. Ця мова створювалася як зручний інструмент для вирішення широкого кола завдань, але зараз C++ виручає там, де потрібна продуктивність і надійність. Наприклад, засобами C++ написані такі успішні продукти [7] на ринку, як: кросплатформний і перший за швидкістю та продуктивністю браузер Google Chrome; дуже вимогливі до продуктивності системи Adobe Creative Suite, що працюють з відео, аудіо та графікою; широко поширена й насичена фічами система Microsoft Office; десктопний клієнт World of Warcraft, який переробляє величезну кількість подій, слідує купі правил, малює графіку і робить ще купу всього пов'язаного з грою. Важко назвати хоча б одну іншу мову програмування, яка дозволяє написати програми подібні до цих і надати такий самий рівень функціональності, стабільності, розширюваності, портабельності. Тому можна з впевненістю стверджувати, що C++ залишатиметься актуальною ще довго.

#### ***Список використаних джерел:***

1. История возникновения языка программирования C++. URL: <http://cppstudio.com/post/1984/>

2. Рейтинг мов програмування 2021. URL: <https://dou.ua/lenta/articles/language-rating-jan-2021/>
3. TIOBE Index for November 2021. URL: <https://www.tiobe.com/tiobe-index/>
4. Що нового у C++20: можливості та перспективи. URL: <https://dou.ua/lenta/columns/c-plus-plus-perspectives/>
5. Talking C++: An Interview with Bjarne Stroustrup. URL: <https://www.codecademy.com/resources/blog/bjarne-stroustrup-interview/>
6. ISO/IEC 14882:2020. Programming languages – C++. URL: <https://www.iso.org/ru/standard/79358.html>.
7. Про ненависть до C++. URL: <https://habr.com/ru/post/111199/>

**Ключові слова:** C++, мова програмування, парадигми програмування.

**Ключевые слова:** C++, язык программирования, парадигмы программирования.

**Keywords:** C++, programming language, programming paradigms.

***Трофименко Олена Григорівна***

*Національний університет «Одеська юридична академія»,  
доцент кафедри інформаційних технологій, кандидат технічних наук,  
доцент*

***Цьоць Артем Олександрович***

*Національний університет «Одеська юридична академія»,  
студент 2-го курсу факультету кібербезпеки та інформаційних*

## **ЩОДО ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ BIG DATA**

Ажіотаж навколо поняття «великі дані» (Big Data), не в останню чергу, зумовлений прагненням компаній та корпорацій отримати якомога більше інформації про потенційних покупців та/або користувачів певної продукції чи