Відкритий міжнародний університет розвитку людини "Україна"

Полтавський інститут економіки і права

Соціально-гуманітарний факультет

Кафедра перекладу та іноземних мов

Допущено до захисту

Завідувач кафедри, кандидат філологічних наук,

доцент\_\_\_\_\_\_\_\_\_\_\_\_\_Рябокінь Н.О.

«\_\_\_\_\_\_» січня 2019 р.

**МАГІСТЕРСЬКА РОБОТА**

на тему:

**«АНГЛОМОВНА ТЕРМІНОЛОГІЯ У ГАЛУЗІ ПРОГРАМУВАННЯ**

**ТА ОСОБЛИВОСТІ ЇЇ ПЕРЕКЛАДУ НА УКРАЇНСЬКУ МОВУ»**

Студента

cоціально-гуманітарного факультету,

спеціальність: 035 "Філологія (Переклад)"

**Келемеша Антона Олександровича**

**Науковий керівник:**

канд. філол. наук,

доцент кафедри перекладу та іноземних мов

**Рябокінь Наталія Олександрівна**

Полтава -2019

2

**ЗМІСТ**

|  |  |
| --- | --- |
| ВСТУП ……………………………………………………………………… | 3 |
| РОЗДІЛ 1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТЕРМІНІВ У ГАЛУЗІ  ПРОГРАМУВАННЯ ……………………………………………………….. | 7 |
| * 1. Визначення понять терміну та термінології у сучасному мовознавстві …………………………………………………………………   2. Історія розвитку та особливості формування терміносистеми у галузі програмування ……………………………………………………….   Висновки до Розділу 1 ……………………………………………………… | 7  16  24 |
| РОЗДІЛ 2. СТРУКТУРНІ ТА СЕМАНТИЧНІ ХАРАКТЕРИСТИКИ ТЕРМІНІВ У ГАЛУЗІ ПРОГРАМУВАННЯ ЯК ЧИННИК ЇХНЬОГО  ПЕРЕКЛАДУ З АНГЛІЙСЬКОЇ МОВИ УКРАЇНСЬКОЮ ……………… | 26 |
| * 1. Терміни як матеріал та як об’єкт дослідження ……………………….   2. Структурний аналіз термінів вибірки …………………………………   3. Семантичний аналіз одиниць вибірки ………………………………...   Висновки до Розділу 2 ……………………………………………………… | 26  30  36  41 |
| РОЗДІЛ 3. АНАЛІЗ ПЕРЕКЛАДУ ТЕРМІНІВ У ГАЛУЗІ  ПРОГРАМУВАННЯ ………………………………………………………...   * 1. Способи перекладу термінів зі словника у галузі програмування …..   2. Способи перекладу термінів у тексті з галузі програмування ………   Висновки до Розділу 3 ……………………………………………………… | 43  43  51  53 |
| ЗАГАЛЬНІ ВИСНОВКИ …………………………………………………… | 54 |
| СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ ……………………………… | 57 |
| ДОДАТОК А. Англо-український глосарій ………………………………. | 62 |

**ВСТУП**

Сучасна лінгвістика активно опановує нові об’єкти дослідження у зв’язку зі зміною гуманітарної ц, зокрема, філологічної парадигми зі структурної на антропоцентричну. Сьогодні, коли у центрі уваги лінгвістів, літературознавців і перекладознавців перебуває людина, яка не тільки користується мовою, а й активно її розбудовує, нашу увагу закономірно привертають різні типи лексичні інновацій, що виникають в різних типах фахової комунікації. Таким чином, **актуальність** дослідження зумовлена дією низки чинників:

* динамічним характером фахової галузі програмування як інноваційного середовища;
* відсутністю комплексних досліджень інноваційної складової фахової галузі програмування з урахуванням як мовних (структурно-семантичних), так і перекладацьких характеристик нових термінів;
* важливістю фахової галузі програмування та його лексичного наповнення для функціонування сучасного «технологічно орієнтованого» суспільства.

**Метою** дослідження є виявлення взаємозв’язку між структурно- семантичною організацією та функціональною сферою вживання термінологічних інновацій, з одного боку, та особливостями їхнього перекладу, з іншого. Реалізація поставленої мети передбачає вирішення низки наступних **завдань**:

* надати визначення понять терміну та термінології у сучасному мовознавстві;
* описати історію розвитку та особливості формування терміносистеми у галузі програмування;
* схарактеризувати терміни як матеріал та як об’єкт дослідження;
* здійснити структурний аналіз термінів вибірки;
* здійснити семантичний аналіз одиниць вибірки;
* визначити способи перекладу термінів у галузі програмування за словником та у контексті.

Отже, **об’єктом** нашого дослідження є терміни фахової галузі програмування, а **предметом** аналізу – їхні структурні, семантичні та перекладацькі характеристики.

**Матеріалом** дослідження виступили 300 англомовних термінів фахової галузі програмування, відібраних зі словників та з тексту «Мова програмування C Розділ 1 Вступний урок» та їхні переклади українською мовою.

**Методи** дослідження зумовлені його метою та завданнями та включають такі: класифікаційний аналіз, жанрово-стилістичний аналіз, структурний аналіз, словотвірний аналіз, семантичний аналіз, порівняльний аналіз одиниць оригіналу та перекладу.

**Положення, що виносяться на захист:**

1. Найвагомішим здобутком для людства другої половини минулого сторіччя став стрімкий зріст в глобальному масштабі комп’ютерної техніки і технологій. Однаково стрімко розвивалися також спеціальна мова і спеціальна термінологія, що обслуговують цю предметну галузь. Комп’ютерна термінологія

– це частина більш загального поняття, а саме, частина спеціальної лексики комп’ютерної мови – одна з новітніх предметних галузей спеціального знання, нарівні з кібернетикою, робототехнікою, біонікою, тобто усі провідні галузі без яких неможна уявити наше майбутнє . Така термінологія багато в чому навіть випереджає спеціальну лексику перерахованих галузей, що помітно за темпами її поширення серед різних соціальних груп різного віку. Тож від її вивчення, можна сказати, залежить успіх нашого майбутнього.

Формалізація і формування комп’ютерних термінів відповідної предметної галузі рухаються в одному напрямку – спілкуванні людини і ЕОМ природною мовою, тож буде логічно припустити, що сфери національної мови та мови комп’ютерної будуть все більш перетинатися. Це необхідно враховувати при аналізі процесів, що відбуваються в комп’ютерній термінології.

1. Для утворення термінологічних одиниць у галузі програмування були застосовані такі способи: словоскладання (одно та багатокомпонентне) – 230 одиниць (76% вибірки); лексико-семантична транспозиція(вторинна номінація) – 12 одиниць (4% вибірки); абревіація – 3 одиниці (1% вибірки); суфіксація – 34 одиниці (11 % вибірки); префіксація – 23 одиниці (7% вибірки);суфіксація + префіксація – 5 одиниць(1,6 % вибірки).
2. Терміни-іменники нашої вибірки представлені наступними трьома тематичними групами: (1) найменування абстрактного поняття; (2) найменування особи; (3) найменування предмету. Група «найменування абстрактного поняття» включає такі ЛСГ: програма / операція; виміри/одиниці обчислення; процес / система. Група «найменування предмету» представлена такими ЛСГ: інструмент/технічна складова/деталь; пристрій/девайс. Група

«найменування особи» представлена ЛСГ «фахівець/майстер».

1. При перекладі одиниць вибірки застосовані такі способи перекладу та трансформації лексичного рівню: Калькування + транскодування (50 одиниць або 16,6%); калькування (30 одиниць або 10%); транскодування (60 одиниць або 20%); вживання еквіваленту з мови перекладу (155 одиниць або 51,6%); описовий переклад (3 одиниці або 1%). Ми визначили що найбільш вживаним засобом є вживання еквіваленту з мови перекладу.

**Теоретичне значення** роботи полягає в тому, що вона запроваджує комплексний підхід до визначення специфіки перекладу термінів, зокрема, термінів-інновацій, за якого спосіб перекладу визначається на основі урахування перекладачем їхньої функціональної сфери, її цільової спрямованості (прагматичного навантаження) та структурно-семантичної організації. Головні положення роботи можуть вважатися внеском у загальну теорію перекладу (в аспекті перекладацьких труднощів), спеціальну теорію науково-технічного перекладу та часткову теорію англо-українського перекладу.

**Практичне значення** роботи полягає в тому, що її методологія та отримані результати можуть стати у принаді викладачам курсів порівняльної лексикології англійської та української мов та вступу до перекладознавства, а також молодим науковцям при проведенні досліджень. На основі вибірки може бути укладений словник-глосарій термінів економічної географії.

**Апробація результатів дослідження***.* Головні результати дослідження викладено у доповіді двох статтях на міжнародних науково-практичних конференціях в 2017-2018 роках.

**РОЗДІЛ 1**

**ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТЕРМІНІВ У ГАЛУЗІ ПРОГРАМУВАННЯ**

* 1. **Визначення понять терміну та термінології у сучасному мовознавстві**

Вивчення проблем термінології надзвичайно актуальне та важливе, тому що терміни відносяться до лексики, яка розвивається швидкими темпами та користується попитом у фахівців різних сфер. Також є важливим те, що термінологія – це головне джерело поповнення лексичного складу високорозвинених сучасних мов.

Початок розробки термінології як науки був закладений незалежно один від одного австрійським науковцем Ойгеном Вюрстером та російським термінознавцем Дмитром Семеновичем Лотте, що опублікували перші термінологічні праці у 1930 році, а сам термін термінологія було запропоновано лише у 60-х роках ХХ ст.

Нині наявність окремої науки про терміни вже не викликає заперечень, більше того в її межах розвиваються самостійні напрямки і школи. Але незважаючи на те що термінологія є однією з найскладніших проблем у галузі лінгвістики та перекладознавство, не всі можуть правильно відповісти на запитання, що таке термін. До того ж, саме поняття термін має відмінне тлумачення у різних джерелах. Наведемо декілька прикладів.

Згідно з он-лайн енциклопедією Вікіпедія, терміном є «[слово](https://uk.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B2%D0%BE) або [словосполучення](https://uk.wikipedia.org/wiki/%D0%A1%D0%BB%D0%BE%D0%B2%D0%BE%D1%81%D0%BF%D0%BE%D0%BB%D1%83%D1%87%D0%B5%D0%BD%D0%BD%D1%8F), яке застосоване для позначення деякого [поняття](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BD%D1%8F%D1%82%D1%82%D1%8F)» [5].

Згідно з гуманітарною енциклопедією терміном є вираз природної чи штучної мови, який має на увазі предмет (реальний або нереальний) чи сукупність предметів [7].

К. Я. Авербух характеризує термін як «складний об’єкт що поднює у собі лінгвістичні та поняттєві початки» [1, с. 20].

О. Б. Бушев визначає термін як «таке слово чи словосполучення, яке означає поняття у спеціальній галузі діяльності чи значення» [4, c. 36].

М. А. Федоров характеризує термін як «сукупність масиву інформації» [20,

c. 42].

О О. Реформатський визначає термін як «слово спеціальне, обмежене

особливим значенням. Метою цього слова є однозначність та точність виразу понять i назв речей» [19, c. 21].

В. C. Кулебакiн вважає терміном «слово чи словосполучення у якого мовний знак спільний з поняттям у системі понять, особливої галузі науки та техніки» [11, c. 80].

Н. Н. Лантюхова характеризує термін як «слово чи словосполучення, спеціальної мови, яке створюється, приймається та запозичується за для точного виразу спеціальних понять чи позначень спеціальних предметів» [12, c. 56].

Таким чином, можна визначити, що для терміна є спільними такі риси: терміном є слово або декілька слів, що найбільш влучно визначають сутність галузевих понять, та назв речей, і є постійним відповідником. І.С. Квітко, спираючись на різні значення, пропонує цікаву дефініцію: «Термін – це слово чи словесний комплекс, який співвідноситься з поняттям певної організованої галузі пізнання (науки, техніки), що вступають у системні відносини з іншими словами та словесними комплексами й утворюють разом з ними в кожному окремому випадку та в певний час замкнену систему, що характеризується високою інформативністю, однозначністю, точністю та експресивною нейтральністю» [9, c. 66].

Однак треба зазначити, що між термінологією і нетермінологією не існує непрохідної прірви, що терміни складаються з однакових звуків і підкоряються однаковим граматичним законами конкретної мови. Якщо б це було не так, то термінологія не належала б до даної мові і взагалі представляла б собою іншу мову. Звідки б терміни не черпалися і якими б особливостями (фонетичними, граматичними) не відрізнялися вони включаються в словниковий склад мови і підкоряються його фонетичному і граматичному строю. Той же самий термін

може входити в різні термінології цієї мови, що являє собою між наукову термінологічну омонімію. Термін *мова,* у програмуванні, – це типовий міжнауковий омонім, а саме його значення «мова програмування» [30 c.33].

Омонімія – це відносини лексичних одиниць, що збігаються за формою і не зв’язані за значенням. На відміну від багатозначності, яка носить регулярний характер і зустрічається у значної частини лексичних одиниць, омонімія – явище в мові порівняно рідкісне, характерне для окремих лексичних утворень. Наприклад, іменник «миша» має два значення: пряме – «тварина», і переносне –

«комп’ютерний пристрій», що регулярно, майже завжди застосовується з комп’ютером. Слова, тотожні за формою і не пов’язані за значенням, називаються омонімами або лексичними омонімами. Лексичні омоніми поділяють на повні та часткові.

Повні лексичні омоніми відносяться до однієї частини мови і мають одну і ту ж систему форм: 1) «откатка» – відновлення минулих варіантів текстового документу у програмуванні і «откатка» – відбракування м’яча з міста штрафного удару у футболі; 2) «трансляція» – переклад програми з мови програмування високого рівня на мову зрозумілу процесору комп’ютера і «трансляція» – передача по радіо чи телебаченню з місця змагань [49 c.165].

Часткові лексичні омоніми можуть належати як до однієї частини мови, так і до різних, але навіть належачи до однієї частини мови, вони мають системи форм, що не збігаються цілком.

Повні та часткові лексичні омоніми являють собою ядро омонімії, навколо якого розташовуються мовні одиниці (омоформи, омоморфеми, омофони і омографи), співзвуччя яких визначається граматичними, словотворчими і фонетичними чинниками.

Хороші терміни повинні бути «відмежовані» [15, c. 12] від полісемії, від експресивності і тим самим від звичайних нетермінологічних слів, які як раз переважно багатозначні й експресивні.

Полісемія (від грець. *poly* – «багато» і *sema* – «знак, значення»), або багатозначність слів визначається як наявність у семантичній структурі мовної

одиниці двох, кількох чи багатьох значень. Однією з причин полісемії у термінології певної галузі є недостатня вивченість і визначеність певних понять, зокрема, нових, що не дає можливості подати їхню чітку й однозначну дефініцію, яка включала б усі їх ознаки, а не лише істотні. Термінологізація відбувається і внаслідок перенесення назви з існуючого предмета чи явища на новий за подібністю (метафора) чи суміжністю (метонімія), які в загальнолітературній мові сприяють створенню образності, а в спеціальній номінації використовуються для утворення нових термінів. Зв’язок зі словом, яке дало ім’я новому терміну, зберігається лише у формі семантичної вмотивованості останнього. Полісемічні слова протиставні моносемічним (від грец. *monos* –

«один» і *sema* – «знак, значення»), або однозначним, якими найчастіше виступають терміни, котрі також нерідко підлягають лексико-семантичним змінам, що безпосередньо пов’язані з формуванням багатозначності слів. Таким чином, існують однозначні та багатозначні терміни. Однозначними (що мають одне лексичне значення) словами є більшість термінів, деякі назви інструментів, професій, різновидів дерев та ін. Однозначними є, наприклад, слова «монітор»,

«комп’ютер», «програміст», «дріт» (див. додаток). Багатозначні (що мають кілька значень) слова мають пряме і переносне значення. Пряме значення – це основне лексичне значення слова. Переносне значення слова – вторинне значення слова, яке виникло на основі прямого. Багатозначні терміни набувають різних синонімів і антонімів відповідно до кожного із значень. Так, іменник

«шлюз» у програмуванні – це засоби що запроваджують зв’язок двох локальних мереж. Як термін у галузі будування «шлюз» – це споруда на річці чи каналі що пропускає кораблі при різному рівні води на шляху.

Постійний обмін відбувається між термінами і нетермінами: термінами стають слова загальної мови, втрачаючи деякі свої властивості, не перестаючи бути фактами спільної мови: «миша» у комп’ютері, «порт» у комп’ютері,

«комп’ютерна шина». Також терміни програмування що, навпаки, входять у спільну мову: «фільтрувати», «гальмувати», «папка». Терміни навіть можуть ставати особливими ідіоматичними виразами: «прошити залізо», «закинути

файл», «програма висне». Коли слово стає терміном, то його значення спеціалізується і обмежується. Залежно від тієї чи іншої термінології, куди потрапляє дане слово, виходить нове значення і звідси інші поєднання з оточуючими словами (визначеннями та доповненнями). У складі термінів існують слова, наявні тільки як терміни і в межах однієї термінології («штриховий код», «байт», «браузер», «домен» тощо); бувають і такі, які існують теж тільки як терміни, але беруть участь у різних термінологіях («операція»,

«алгоритм», «аргумент», «масив», «кнопка», «константа» тощо); бувають (що найчастіше) і такі, які вживаються і як терміни, і як звичайні нетерміни, наприклад при відмінності прямого та переносного (метафоричного) значення:

«мишка» – «маленька миша», «мишка» – як термін комп'ютерної справи.

Є декілька шляхів потрапляння термінів до мови. По-перше – це вживання в якості терміна якогось слова загальновживаної мови. Цей шлях може вважатися позитивним, тому що мова зберігає свій словниковий склад, не засмічується іншомовними словами і слово, використане як термін вважається загальнозрозумілим будь-якому хто говорить цією мовою. З іншої сторони, тут є також i мінуси. Насамперед, слово як термін має інше значення, метафоричне, а в іншому випадку і метонімічне, що ще важче розгледіти, виходячи з основного і прямого значення. Так що, розуміючи значення звичайного слова, не завжди легко зрозуміти значення терміна. Для уникання цих неясносте термінологія часто «відокремлює» [15, c. 61] терміни від звичайних слів фонетично чи граматично. Інша можливість «відмежування» терміна – морфологічна.

Зазначених труднощів уникають терміни, які є запозиченнями з інших мов. Вони входять у мову майже як власні імена («комп’ютер», «константа», «код» та інші). Як назви речей і явищ («гігагерц», «інтернет», «інтранет» тощо) вони при ізольованому вживанні чисто номінативні і отримують семасіологічну функцію тільки після утворення похідних термінів, коли спільність поняття виступає як сполучна різних номінацій різних слів. Такі запозичені терміни не змішуються зі звичайними словами. І таких випадків багато в будь якій термінології. Вибір мови-джерела цих термінів обумовлений реальною історичною практикою, тому

тут дуже ясно можна показати зв'язок народів, націй і характер їх культурної взаємодії. В українській мові комп’ютерна термінологія переважно складається з англійських слів, («аргумент», «байт», «код», «елемент», «гігабайт», «Інтранет», «лептоп», «дисплей», «мегабайт», «логін» тощо).

У запозиченні слів є й недоліки, вони полягають у тому, що мова засмічується випадковим чужемовленням, що позначається, насамперед, на строкатості самої термінології і на розриві міжнародних зв’язків, адже різні народи можуть запозичувати термінологію тієї ж галузі з різних мов.

Крім того, при усному побутовому засвоєнні термінів виникають спотворення слів на ґрунті народної етимології.

Існують також випадкові запозичення з окремих мов, які можна проілюструвати наступним прикладом. Термін «коментар» утворений від англійського субстантивованого дієслова *to comment –* «коментувати».

Завдяки зазначеним властивостям, термінологію в будь-якій мові можна і треба впорядковувати, приводити в єдність синонімію, уточнювати значення, уніфікувати форму термінів.

Н. В. Юшманов виводить таку формулу впорядкованого терміна:

«Упорядкована термінологія повинна дати двосторонню відповідність: якщо знаєш термін, знаєш місце в системі, якщо тобі відомо місце у системі, знаєш термін» [23, c. 71].

Сам термін «термінологія» побудований зі змішаних латино-грецьких елементів: лат. *terminus* – «межа», *logos –* «наука».

На сучасному етапі розвитку перекладознавства підвищена увага приділяється оптимізації перекладу термінологічних одиниць фахових мов.

Як вважають прихильники субстанційного погляду, терміни – це особливі слова чи словосполучення, які відрізняються від інших номінативних одиниць однозначністю, точністю, системністю та незалежністю від контексту [5, с. 11]. Але більшість термінів не мають перелічених ознак, тому значна частина мовознавців цей підхід, який протиставляє слово і термін, вважає «відхиленим сучасною наукою» [7, с. 28; 3, с. 10]. Згідно з функціональним (описовим,

дескриптивним) підходом [4, с. 17], терміни – це вже не особливі слова, а слова

«в особливій функції». За цим підходом протиставлення «термін – слово» виявилося вкрай продуктивним за для створення теорії терміну.

Однак більшість цих «особливих» функцій притаманна також загальновживаним словам. До цієї теми доречно навести міркування О.О. Реформатського: «З одного боку, галузь термінології є замкнутою, а з іншого – перебуває у постійній взаємодії з повсякденним мовленням. Кожне звичайне неслужбове слово може стати терміном якщо буде включене в особливий словник за принципом точної відповідності з певною соціально організованою річчю» [19, c. 129]. Також, «чіткої межі між термінами й загальновживаною лексикою не існує. Між загальнонародною мовою та термінологією не завершується процес взаємообміну форми термінологізації й детермінологізації» [35, с. 173].

Один із найвідоміших представників австрійсько-німецької термінологічної школи, автор відомого навчального посібника з термінології

X. Фельбер відокремлює термін за контекстом логіко-лінгвістичного підходу:

«Термін є умовним символом, словом або групою слів, яка виражає конкретне поняття в певній галузі знання» [24]. Із логіко-лінгвістичної співвідносності спеціального поняття з мовною одиницею виходить провідний представник польської термінологічної школи Л. Бесекірська: термін розглядається нею як «слово чи словосполучення, що виражене в змістовому плані технічним, науковим чи іншим спеціальним поняттям» [3].

Розглянувши ці дефініції, ми дійшли висновку що вони є найвдалішими серед тих,що ми проаналізували. До особливостей терміну можна також додати, що він має цілком певне значення і логічно виділяє достатні для вираження та необхідні особливості будь якого поняття чи ознаки, на основі яких вибудовується класифікація понять. В. М. Лейчик і С. Д. Шелов уточнювали, що терміни фігурують як ті ж самі в спеціальній лексиці мови (лексиці для спеціальних цілей), а не в лексиці тієї чи іншої природної мови в цілому [13; 21].

Таким чином ми можемо зазначити що, спеціальною є саме сфера застосування терміну, а не обов’язково термінолексема.

В результаті вивчення різних областей спеціальної лексики було встановлено, що поряд з термінами існують й інші спеціальні лексичні одиниці, які було виділено та описано: «прототерміни», «терміноїди» та «передтерміни»,

«професіоналізми» та «професійні арготизми». Також досі ще не було вироблено загальновизнаних обґрунтувань для розрізнення термінів і номенів, однак велика кількість термінологів схиляється до того, що перші позначають поняття, а другі

одиничні об’єкти. Ці спеціальні лексеми мають ряд спільних ознак з термінами але мають і відмінності.

Передтерміни – це спеціальні лексеми, що використовуються як терміни для називання нових сформованих понять, але не відповідають основним вимогам, що пред’являються до терміну. В якості передтерміна зазвичай виступають:

а) описовий зворот – багатослівне номінативне словосполучення, яке використовується для називання поняття і дозволяє точно описати його сутність, але не відповідає вимогам стислості;

б) складне словосполучення;

в) поєднання слів, що містить прикметниковий або дієприкметниковий зворот.

Передтерміни використовуються в якості термінів для іменування нових понять, для яких відразу не вдається підібрати відповідні терміни. Від термінів передтерміни відрізняються тимчасовим характером, нестійкістю форми, невиконанням вимогливості і загальноприйнятості, часто і відсутністю стилістичної нейтральності. У більшості випадків згодом предтерміни витісняються термінами. У ряді випадків заміна передтерміна лексичною одиницею, більш відповідною термінологічним вимогам, затягується, і передтермін закріплюється у спеціальній лексиці, набуваючи стійкого характеру.

Досить складним є статус професіоналізмів, які деякі фахівці: а) ототожнюють з термінами;

б) відносять до одиниць ремісничої лексики;

в) відносять до спеціальної лексики неномінативного характеру (дієслова, прислівники, прикметники);

г) відносять до ненормативної спеціальної лексики, яка є обмеженою вживанням в усному мовленні професіоналів в неформальній ситуації, і часто має емоційно-експресивні конотації. Різновидом професіоналізмів є професійні жаргонізми, які не здатні набути нормативного характеру, і їх умовність ясно відчувається мовцями.

Терміноїди – це спеціальні лексеми, що використовуються для назв недостатньо усталених (понять, що формуються) і понять що неоднозначно розуміються, які не мають чітких меж, а отже, й дефініцій. Тому терміноїди не мають таких термінологічних властивостей, як точність значення, контекстуальна незалежність і стійкий характер, хоча і називають поняття.

Прототерміни – це спеціальні лексеми, що з’явилися і застосувалися до формальної появи наук, і тому вони називають не поняття (які виникають з появою науки), а спеціальні уявлення. Прототерміни з тих пір не зникли – вони збереглися в ремісничій лексиці, що дійшла до нас (оскільки відтоді багато спеціальних уявлень увійшли до загального вжитку). З часом, з появою наукових дисциплін, в яких спеціальні предметні уявлення ремесел та деяких інших видів діяльності теоретично осмислюються і перетворюються на системи наукових понять, частина стійко закріплених в спеціальній мові прототермінів включається до наукової термінології, а інші існують у вигляді загальновживаної лексики предметних областей, в яких відсутні (чи ще не сформувалися) науково теоретичні основи, або ж функціонують у вигляді так званих «народних термінологій» [10, c. 55], які використовуються паралельно з науковими термінами, але без зв’язку з понятійною системою. Таким чином, більшість базових термінів старих термінологій колись були прототермінами і зберегли ряд своїх рис – використання для мотивації випадкових, поверхневих ознак або відсутність втрати вмотивованості.

* 1. **Історія розвитку та особливості формування терміносистеми у галузі програмування**

Проблема адекватного перекладу і сприйняття термінів у галузі програмування набуває все більш широких масштабів як для користувачів, так і для розробників програм.

Терміни протягом тривалого часу закріплялись та змінювались у мові паралельно з розвитком та винаходом комп’ютерних програм, тим самим маючи все більший вплив на мову та на суспільство що її використовує. Це є причиною доцільності дослідження історії формування терміносистеми у цій галузі.

Одним із феноменів другої половини минулого сторіччя стало стрімке поширення в глобальному масштабі комп’ютерної техніки і технологій [31,

c. 53]. Також стрімко розвивалися спеціальна мова і спеціальна термінологія, що обслуговують цю предметну галузь. Комп’ютерна термінологія – це частина більш загального поняття, а саме, частина спеціальної лексики комп’ютерної мови – одна з новітніх предметних галузей спеціального знання, нарівні з кібернетикою, робототехнікою, біонікою тощо. Ця термінологія багато в чому навіть випереджає спеціальну лексику перерахованих галузей, що помітно за темпами її поширення серед різних соціальних груп різного віку.

Формування комп’ютерних термінів і формалізація відповідної предметної галузі рухаються в одному напрямку – спілкуванні людини і ЕОМ природною мовою, тому логічно припустити, що сфери національної мови та професійних комп’ютерних найменувань будуть все більш перетинатися [46, c. 33]. Це необхідно враховувати при аналізі процесів, що відбуваються в комп’ютерній термінології.

На зорі комп’ютерної ери машинний код був єдиним засобом спілкування людини з комп’ютером. Величезним досягненням творців мов програмування було те, що вони зуміли змусити сам комп'ютер працювати перекладачем з цих мов на машинний код.

Наприкінці 40-х років, до приходу Грейс Хоппер до фірми Джона Моучлі, останній створив систему під назвою *Short Code*, яка була примітивною мовою програмування високого рівня. У ній програміст записував завдання у вигляді математичних формул, а потім, використовуючи спеціальну таблицю, перекладав символ за символом, перетворюючи ці формули на двійкові коди [42,

c. 22]. Надалі спеціальна програма комп’ютера перетворювала ці коди в двійковий машинний код. Система, розроблена Дж. Моучлі, була по суті одним з перших примітивних інтерпретаторів [26, c. 284].

У 1958 р з’явився компілятор FLOW-MATIC. На відміну від ФОР-Трану – мови для наукових додатків – FLOW-MATIC був першою мовою для обробки комерційних даних. Роботи в цьому напрямку призвели до створення мови COBOL (COBOL – *Common Business Oriented Language*). Одним із основних консультантів при створенні цієї мови була Грейс Мюррей Хоппер [50, с. 28].

Середина 50-х років характеризується стрімким прогресом у галузі програмування. Роль програмування у машинних командах стала зменшуватися. Почали з’являтися мови програмування нового типу, які виступають в ролі посередника між машинами і програмістами. Першою і однією з найбільш поширених була мова Фортран (*FORTRAN*, від *FORmula TRANslator* –

«перекладач формул»), розроблена групою програмістів фірми IBM в 1954 р (перша версія) [33, c. 70].

На початку 60-х років всі існуючі мови програмування високого рівня можна було перерахувати по пальцях, проте згодом їх число досягло трьох тисяч. Зрозуміло, що переважна частина мов не отримала скільки-небудь широкого поширення; у практичній діяльності використовується не більше двох десятків. Розробники орієнтували мови на різні класи задач, тією чи іншою мірою прив’язували їх до конкретних архітектур ЕОМ, реалізовували особисті смаки та ідеї.

У 60-ті роки були зроблені спроби подолати це «різноголосся» шляхом створення універсальної мови програмування. Першим дітищем цього напрямку

стала мова PL / I (*Program Language One*), створена 1967 р. Потім на цю роль претендував АЛГОЛ-68 (1968 р.) [35, c. 52].

Передбачалося, що подібні мови будуть розвиватися і вдосконалюватися та витіснять усі інші. Однак жодна з цих спроб на сьогоднішній день не увінчалася успіхом (хоча PL / I в усічених версіях використовували багато програмістів) [32, c. 148]. Всеосяжність мови приводила до невиправданої, з точки зору програміста, складності конструкцій, неефективності компіляторів. Мови програмування служать різним цілям, а їхній вибір визначається зручністю користувача, придатністю для конкретного комп’ютера і конкретного завдання. А завдання для комп’ютера бувають найрізноманітнішими: обчислювальними, економічними, графічними, експертними тощо. Така різнотипність розв’язуваних комп’ютером завдань визначає різноманіття мов програмування. Вочевидь, в програмуванні найкращий результат досягається при індивідуальному підході, що виходить з. класу завдання, рівня та інтересів програміста. Наприклад, мова Бейзік широко вживається при написанні простих програм [47, c. 75].

Фортран є класичною мовою програмування при вирішенні на ЕОМ математичних та інженерних завдань; мова Кобол (COBOL, від *Common Business Oriented Language* – загальна мова, орієнтована на ділові завдання; створена в 1960 р.) була задумана як основна мова для масової обробки даних у сферах управління та бізнесу [45, c. 81]. Ще більш спеціалізованою є мова ЛОГО (від грецького *logos* – слово), створена для навчання програмуванню школярів професором математики і педагогіки Сеймуром Пейпертом з Масачусетського технологічного інституту. Навчаючись програмуванню на основі ЛОГО, діти задають прості команди, які керують іграшкової черепашкою, забезпеченою олівцем. Відзначимо і ще одну досить популярну спеціалізовану мову Пролог (*Prolog* – *PROgramming* in *LOGic*), розроблену як мову програмування для створення систем штучного інтелекту [50, c. 160].

З середини 70-х рр. в США, в рамках міждисциплінарного підходу, були висунуті припущення про те, що пізнавальні процеси у свідомості людини

аналогічні обчислювальним алгоритмам і що у свідомості людини знаходяться ментальні репрезентації, аналогічні комп’ютерним структурам даних (пропозиції, схеми, поняття, правила). Згідно з цими положеннями про співвідношення природного та штучного інтелекту, було заявлено, що людський мозок за аналогією з комп’ютером управляє центральною і периферичною нервовими системами під впливом зовнішніх обставин. Таким чином, комп’ютерні терміни виступають як нові одиниці номінації в перекроюванні картини світу і заявляють про себе з позиції транснаціональних проблем. Рішення проблеми статусу комп’ютерної мови по відношенню до національної системи дозволить на основі об’єднання природничо-наукових і гуманітарних знань створити несуперечливу картину світ а також, скорегувати наукові висновки і припущення.

Слід зазначити, що багато мов, спочатку розроблених для великих і малих ЕОМ, в подальшому були пристосовані до потреб персональних комп’ютерів. Протягом багатьох років програмне забезпечення будувалося на основі операціональних і процедурних мов, таких як Фортран, Бейзік, Паскаль, Ада, Сі. І сьогодні сучасні версії цих та подібних мов (Модула, Форт і інші) домінують при розробці прикладних програмних засобів [27 c. 91]. Однак внаслідок еволюції мов програмування набули широкого поширення й принципово інші підходи до створення програм.

Класичне операційне або процедурне програмування вимагає від програміста детального опису того, як вирішувати задачу, тобто формулювання алгоритму і його спеціального запису. При цьому очікувані властивості результату зазвичай не вказуються. Основні поняття мов цих груп – оператор і дані. При процедурному підході оператори об’єднуються у групи – процедури. Структурне програмування в цілому не виходить за рамки цього напрямку, воно лише додатково фіксує деякі корисні прийоми технології програмування.

Принципово інший напрямок в програмуванні пов’язаний з методологіями (іноді говорять «парадигмами» [11, c. 15]) непроцедурного програмування. До них можна віднести об’єктно-орієнтоване і декларативне програмування.

Об’єктно-орієнтована мова створює оточення у вигляді безлічі незалежних об’єктів. Кожен об’єкт поводиться подібно окремому комп’ютеру, їх можна використовувати для вирішення завдань як «чорні скриньки», не вникаючи у внутрішні механізми їхнього функціонування. З мов об’єктного програмування, популярних серед професіоналів, слід назвати перш за все *Сі* ++, для більш широкого кола програмістів переважні середовища типу *Delphi* та *Visual Basic* [36, c. 77].

Велика увага в наш час приділяється проблемі створення

«інтелектуальних» ППЗ. Такий пакет дозволяє кінцевому користувачу лише сформулювати своє завдання у змістовних термінах, не вказуючи алгоритм її рішення. Синтез рішення і збірка цільової програми проводяться автоматично.

Зараз є також важливим дослідження принципового формування комп’ютерної термінології та виявлення принципової внутрішньої структури організації і розвитку терміносистеми якоїсь спеціальної підмови.

За рамками національної мови, якою є українська мова, знаходяться:

1. окремі форми комп’ютерних термінів, що не є характерними для української фонологічної та морфологічної структури мови: «картридж»,

«бенчмарк», «браузинг», «драйвер», «дескриптор»;

1. комбінація української та англійської графіки та орфографії, що відсутні в національній мові: «*SIM*-карта», «*Web*-сайт», «*Р*-телефонія», «*Web*-сторінка», у мережі – «*WWW*»;
2. інтернаціональні гібридні словосполучення термінів, що виражають атрибутивні відношення, які є характерними для англійської мови: «смарт-тег»,

«файл-сервер», «клієнт-сервер», «компакт-диск», «майстер-диск», «флоппі- диск».

Таким чином, відбувається більшість перекладів термінів: *power* – «блок питання»; *model* – «модель», «моделювати»; *printer* – «печатний пристрій», *tester*

* «випробувальний пристрій».

Неспіввідносність змісту українських та англійських комп’ютерних термінів може мати місце за рахунок:

1. неспіввідносності лексичних значень у відповідних поняттях: (RОМ (пам’ять лише для читання) – «постійна пам’ять»);
2. неспіввідносності обсягу понять: (*integrated circuit* – «елементна база»,

*speed* – «швидкісна база»);

1. неспіввідносності обсягу засобів вираження понять (у полісемії та синонімії): *data flow, data stream* – «рух даних»; *scroll bar* – «лінія прокрутки»,

«прокрутка стрічки»;

1. одному недиференційованому поняттю в одній мові відповідає декілька понять в іншій: *World Wide Web* – «всесвітня павутина», «інтернет»;
2. поняття повністю відсутнє у взаємодіючій мові, тоді займається форма та зміст даного поняття (*web*-сайт, Інтернет та інші.).

Останні комп’ютерні терміни складають значний рівень безеквівалентної лексики у комп’ютерній термінології. Слова з повсякденного вжитку виловлюються з словників і їх значення перекроюються відповідно до реалій комп'ютерного світу.

Розглянемо декілька з них.

1. «Миша» – це навігаційний пристрій, що використовується для комп’ютерів, походження терміна невідомо. Навіть винахідник пристрою Дуглас Енгельбарт має лише здогади з цього приводу. У своєму інтерв’ю він сказав, що ніхто не міг згадати походження цього терміна, лише те що пристрій виглядав як миша з хвостом, і він з командою так і називали його під час роботи в лабораторії. Апаратний дизайнер Роджер Бейтс, який теж у той час працював над розробкою миші, по-іншому тлумачить походження назви. Він пише, що курсор на екрані комп’ютера назвали *CAT* («Кішка»), а кішки, як ми знаємо, полюють

на мишей [43, с. 237].

1. «Блог» – це персональний сайт для написання повідомлень та обміну посиланнями. Термін насправді є скороченням від прізвиська «веблог» (*weblog*). Був створений у 1997 році Йорном Баргером. Термін відносять до його веб- сторінки *Robot Wisdom*, де автор коментував свої Інтернет поневіряння. З плином

часу слово було все більше використано, і їм стали називати популярний час проведення у мережі [25, c. 141].

1. «Кукіс» – це невеликий фрагмент інформації що зберігається при відвідуванні веб-сторінки. Він походить від «чарівного печива» (*Magic cookie*) – старого комп’ютерного терміна з практично тим же значенням. Лу Монтуллі, винахідник веб кукис, у своєму блозі пояснив вибір даних слів. Він чув про термін «чарівне печиво» з курсу по операційним системам в коледжі. Йому сподобався термін «кукіз» з естетичних міркувань. Це було перше, що він придумав, і назва прижилася. Що стосується більш глибокого коріння терміна, то деякі теорії посилаються на старі відеоігри, де гравці отримували «чарівні печива» в цілях просування [5].
2. «Спам» – це кількість небажаної пошти. Термін походить з англійського шоу «Монті Пайтон». У старому комедійному скетчі показували обід зі СПАМом (дешеві м'ясні консерви) в кожній страві. Незабаром герої скетчу виспівували і скандували слово «спам» по кілька разів. Термін прижився в Інтернет-чатах і його стали асоціювати з дратівливими, повторюваними небажаними речами.
3. «Хакер» – це комп’ютерний злочинець, який здійснює злом і несанкціонований доступ до даних. Цей термін не завжди був негативним явищем. На початку технологічної ери хакерами називали розумних і талановитих геніїв техніки, необов’язково комп’ютерної. У книзі «Піратські культури» йдеться про походження терміна від фрази «той, хто працює як кінь» в написанні програм і експериментує з ними. Позитивні хакери все ще існують. Представники цієї культури вважають за краще називати злісних хакерів

«крекерами».

1. «Фаєрвол» – це програма для захисту комп’ютера від шкідливих хакерів та вірусів. Термін існував протягом сотень років і означав саме те, що і повинен був («фаєрвол» англійською «вогняна стіна»). Так у стародавні часи називали стіни, які захищали будівлі від поширення пожежі в місті. Функції комп’ютерної версії схожі: вона захищає від поширення шкідливих вірусів.

Таким чином ми розглянули повсякденну термінологію, але також треба звернути увагу на терміни з що також впливають на мову та термінологію – спеціальні терміни. Наведемо декілька прикладів що вживаються програмістами за для обговорення програм та мові програмування в усній і письмовій формі.

1. «Ключове слово» (*keyword*) – слово, яке є частиною мови. Ключові слова неприпустимі в якості ідентифікаторів Приклади: *function*, *if*, *while*, *var*.
2. «Зарезервоване слово» (*reserved word*) – слово, яке не рекомендується використовувати в якості ідентифікатора. Зарезервовані слова можуть виявитися ключовими в більш пізніх версіях мови.
3. «Оператор» (*statement*) – команда, до виконання певної дії комп’ютером; оператори зазвичай призводять до зміни стану оточення (наприклад, змінної або визначення). Приклади: *х = х + 5, function myfunc (x, у) {return x \* y}*.
4. «Літерал» (*literal*) – значення, що міститься безпосередньо в тексті програми. Приклади: 5, «Привіт всім!», *false* [3; 7; 9; 25].
5. «Лексема» (*token*) – найменша і неподільна одиниця мови. Приклади: всі ідентифікатори, ключові слова, а також літерали типу 5.2 і «Привіт всім!».
6. «Знак операції» (*operator*) – лексеми, що представляють собою вбудовані оператори мови. Приклади: =, +, -, \*, / (оператор присвоювання і арифметичні оператори). Операції нерідко називають операторами.
7. «Вираз» (*expression*) – група лексем (зазвичай літералів і ідентифікаторів) в поєднанні зі знаками операцій, для яких можна обчислити значення. Приклади: 3.141, «Привіт!», (2 + х) \* 5, *myfunc* (), x = у + *myfunc* ().
8. «Реєстрова залежність» (*registry addiction*) – залежність виразів мови або їх частин (наприклад, імен змінних, функцій та інші.) Наприклад, мова *JavaScript* повністю чутлива до регістру, а РНР – теж чутливі до регістру, але не повною мірою.
9. «Ідентифікатор» (*identifier*) – ім’я змінної, функції, об'єкта та ін. Ідентифікатор не повинен бути ключовим або зарезервованим словом. Програміст зазвичай вибирає ідентифікатори на свій розсуд, але відповідно до певних правил. У більшості випадків ідентифікатори складаються з друкованих

символів (букв, цифр і символу підкреслення), причому ідентифікатор не повинен починатися з цифри і містити символи пробілу. Приклади: *х, myvar, userName, іsеr\_name, var12*.

**Висновки до Розділу 1**

1. Спираючись на дослідження науковців, наведені у цьому розділі, та керуючись власною думкою, ми визначили поняття терміну як слова або словосполучення, що найбільш влучно визначає сутність галузевих понять, та назв речей, а в аспекті перекладу характеризується наявністю постійного відповідника.
2. Аналізуючи теоретичні джерела у галузі програмування та наведені приклади в галузі термінології програмування, можна зробити висновок, що першорядними труднощами у процесі перекладу досліджуваних одиниць є такі фактори як: наявність абревіатур, скорочень в даній галузі; потреби враховувати особливості перекладу складних граматичних конструкцій; адекватність вибору правильного значення полісемантичного слова; велика кількість безеквівалентних лексичних одиниць у складі комп’ютерної термінології англійської мови; відсутність єдиної дійсної моделі перекладу комп’ютерних текстів; морфологічна будова і семантичні особливості термінів; постійне оновлення лексики і поява комп’ютерних неологізмів. Окремим пунктом виступає обов’язкова індивідуальна грамотність перекладача, що є ключовим аспектом успішної комунікації.
3. Слід також зазначити, що абсолютно всі перераховані вище особливості перекладу англійської комп’ютерної лексики українською мовою мають надзвичайно вагоме значення у сфері міжкультурного спілкування. Правильний підхід до труднощів у процесі перекладу комп’ютерної лексики гарантує перекладачеві успішний результат у сприйманні та наданні необхідної інформації.

**РОЗДІЛ 2**

**СТРУКТУРНІ ТА СЕМАНТИЧНІ ХАРАКТЕРИСТИКИ ТЕРМІНІВ У ГАЛУЗІ ПРОГРАМУВАННЯ ЯК ЧИННИК ЇХНЬОГО ПЕРЕКЛАДУ**

**З АНГЛІЙСЬКОЇ МОВИ УКРАЇНСЬКОЮ**

* 1. **Терміни як матеріал та як об’єкт дослідження**

За матеріал дослідження ми обрали 300 термінів у галузі програмування та їхні переклади українською мовою. Програмування є відносно молодою науковою дисципліною. Отже, для аналізу ми обирали терміни, що були створені на початку розвитку програмування і ті що були засновані відносно недавно.

Такий широкий часовий діапазон визначення «інновацій» (зазвичай такими вважаються неологізми останніх 5 – 10 років) виправданий тим, що термінологічний апарат програмування українською мовою залишається практично не сформованим. Таким чином, фактично усі переклади нам довелося виконувати самостійно, спираючись як на існуючі (в деяких випадках) російські еквіваленти, так і на англомовні дефініції глосаріїв.

Оскільки лексичні одиниці, що є матеріалом та об’єктом нашого аналізу, є термінами у складі термінологічної системи програмування, маємо з’ясувати, в якому значенні слова «термін» та «термінологія» вживаються у цій роботі.

Англійська терміносистема програмування є однією з найбільш розвинених та упорядкованих сучасних систем термінів, яка постійно змінюється та поповнюється новими одиницями. Проблема її вивчення та комплексного аналізу є особливо актуальною сьогодні, оскільки англійська мова перетворилася на єдиний глобальний мовний код та набула статусу мови міжнародних наукових зв’язків. Саме цьому знання сучасної англійської термінолексики необхідне при проведенні міжнародних наукових конференцій, симпозіумів та диспутів з тематики програмування.

Знання структурних, семантичних та функціональних особливостей сучасної англійської термінолексики програмування необхідні для підготовки

наукових та технічних кадрів, видання літератури у галузі програмування, оформлення програмної документації тощо.

Серед проблем, що постійно перебувають у центрі уваги лінгвістів, є проблеми, пов’язані з динамічними мовними явищами, які відображають розвиток суспільства. Саме до таких належать дослідження, присвячені вивченню закономірностей становлення та розвитку термінологій, що обслуговують певну галузь людської діяльності, в нашому випадку програмування.

Вперше нормативні вимоги до терміна (тобто, вимоги на рівні системності науково-лінгвістичного аналізу) були сформульовані основоположником російської термінологічної школи Д. С. Лотте. Відповідно до запропонованого ним підходу, під терміном розуміється «слово або стійке сполучення, що служить уточненим найменуванням поняття, специфічного для певної галузі знання, виробництва, культури» [14, с. 25].

На сучасному етапі розвитку термінознавства окреслилося два основних підходи у дослідженні термінів – нормативний та дескриптивний. До представників нормативного підходу відноситься, зокрема, О.С. Ахманова. Термін розглядається нею як «слово (або словосполучення) спеціальної сфери вживання, яке є найменуванням спеціального поняття та вимагає дефініції» [2, с. 532]. Таким чином, термін за своєю структурою та значенням протиставлений загальновживаним словам чи словосполученням.

Дескриптивний підхід ґрунтується на положенні Г.О. Винокура про те, що

«терміном може виступати будь-яке слово, оскільки …терміни – це не особливі слова, а лише слова у особливій функції» [6, с. 5]. При такому підході специфіка терміна полягає у його високому рівні інформативності, вмотивованості та узагальнення, в особливому призначенні, функціях, насамперед, функції називання, позначення спеціальних понять, наукової комунікації.

У своїй роботі, слідом за представниками дескриптивного підходу, ми визначаємо термін як «одиницю найменування певної області науки та техніки, створену штучно чи взяту з природної мови, якій приписується певне поняття і

яка співвідноситься з іншими найменуваннями у цій галузі та утворює спільно з ними термінологічну систему» [6, с. 15].

Головна тенденція розвитку термінологічної номінації полягає у складному процесі взаємодії загальнолітературної мови і її функціональних різновидів (спеціальних дискурсів). Усе, що з’являється у сфері професійної або наукової комунікації, «реалізується засобами загальнолітературної мови, одержуючи певну модифікацію, а потім повертається до неї як стимул для розвитку і поширення явищ, що виникають у її підсистемах» [6, с. 47]. Таким чином, використання загальновживаних слів у функції термінів випливає з узагальнюючої природи слова і тих потенційних можливостей, що закладені в його семантиці. Значення багатьох слів постійно розширюються і збагачуються новими вузькоспеціальними ознаками, внаслідок чого ці слова переходять у категорію термінів.

Проблема розмежування термінів та слів загальновживаної мови є однією з найважливіших проблем теорії термінології. Термін володіє низкою характеристик, які відрізняють його від слів загальнолітературної мови. Разом із Г. О. Винокуром головними його ознаками ми вважаємо такі:

1. *специфічність*, яка проявляється у чіткому відмежовані термінів за сферою використання;
2. *дефінітивність*, яка проявляється у наявності чітко окресленої дефініції

* логічно обґрунтованого визначення поняття на основі його характерних інтегральних та диференціальних ознак;

1. *системність*, яка проявляється у тому, що кожен термін має строго визначене місце у термінологічній системі, яке залежить від місця відповідного поняття у загальній системі галузевих понять;
2. *однозначність*, яка проявляється у тому, що термін, на відміну від звичайного слова, виражає обмежене, твердо фіксоване поняття, тобто він є однозначним. Коментуючи цю ознаку зазначимо, що вона сьогодні ставиться під сумнів численними дослідженнями, які свідчать про багатозначність багатьох

термінів. Робимо висновок: терміни лише мають тенденцію до однозначності у межах власної термінологічної системи;

1. *стилістична нейтральність*, яка проявляється у відсутності конотативніх смислів експресивного, емоційного чи аксіологічного характеру;
2. *інтернаціональність*, яка проявляється у зрозумілості (принаймні найважливіших, базових) термінів спеціалістами певної галузі в різних країнах. Саме через це при перекладі термінів одним із найчастотніших є спосіб транскодування, який забезпечує упізнаність таких одиниць навіть за умови зміни абетки (тобто, при перекладі, наприклад, з української мови англійською або навпаки);

9) *зручність*, яка проявляється у легкій відтворюваності терміна. За сучасних умов ця вимога часто порушується, оскільки високий рівень наукової спеціалізації призводить до все більшого утворення багатокомпонентних термінологічних словосполучень з великою кількістю компонентів. Такі формації не тільки складно відтворювати, а й не менш складно перестворювати за умов перекладу. «Існують навіть спеціальні рекомендації, які допомагають перекладачам зорієнтуватися з порядком відтворення складових багатокомпонентного терміну» [6].

Сукупність перерахованих ознак існує лише в ідеалі для невеликої кількості термінів. У дійсності ж та чи інша ознака або відсутня, або існує в послабленому стані. Саме тому у термінознавстві існує таке поняття, як «ідеальний термін» [6, с. 34].

Певні суперечки серед фахівців викликає питання частиномовної приналежності терміна. Поширеною є думка, що термінами є лише іменники, оскільки лише вони мають номінативний характер. Інші припускають, що термінами можуть бути і слова, що належать до різних частин мови, насамперед, до повнозначних. Д. С. Лотте виділяв чотири основні категорії термінів, які слугують потребам номінації предметів (іменники), процесів / явищ (дієслова), якостей (прикметники), величин (прислівники) [14]. Визнаючи зв’язок між системою субмови науки і мовною системою в цілому, сьогодні у

термінознавчих дослідженнях поширеною є позиція, згідно з якою термінами можуть бути слова, представлені різними частинами мови, знаменними насамперед, якщо у складі визначеної термінології вони набули термінологічного змісту і служать для вираження системних відносин.

Термінологічна номінація – це усвідомлений, цілеспрямований творчий процес. Це свідчить про те, що терміни побудовані більш строго і більшою мірою спираються на закони формальної логіки. Як бачимо, терміни і терміносистеми перебувають на стику природного і штучного у мові, при цьому «природним є переважно мовний матеріал, а штучним – використання цього матеріалу» [14, с. 11].

Враховуючи усі зазначені вище міркування стосовно характеристичних ознак терміна, у своїй роботі ми дотримуємося точки зору, згідно з якою терміном може бути слово будь-якої частини мови або словосполучення, що виражає спеціальне поняття, має точну дефініцію, виконує функції знакової систематизації, наукової комунікації, гносеологічну та евристичну функції, характеризується точністю, стислістю та легкістю творення похідних термінів. Це визначення має для нас передусім прикладне значення, оскільки дозволяє встановити межу між термінами та не-термінами у процесі відбору ілюстративного матеріалу та визначає пріоритети їхнього іншомовного відтворення, яке має забезпечити у відповідників наявність термінологічних ознак, виокремлених в оригіналі.

* 1. **Структурний аналіз термінів вибірки**

Усі терміносистеми є саморегульованими, вони реагують на зовнішні зміни через надбання або, навпаки, втрату своїх елементів, залучаючи найбільш оптимальні для цього способи. Таким чином, вивчення структурних характеристик термінів інновацій може бути корисним з точки зору з’ясування:

1) продуктивних / малопродуктивних / непродуктивних способів / засобів

номінації термінотворення; 2) ступеню складності концептуального апарату та понять даної галузі (на основі сполучення структурного аналізу з семантичним). З точки зору транслятології, структура терміну розглядається як надзвичайно важливий чинник перекладу, оскільки для досягнення максимальної адекватності перекладач має відтворити структуру вихідної

одиниці у цільовій мові.

З обраних трьохсот термінів для утворення термінологічних одиниць у галузі програмування було застосовано лише декілька способів, продуктивність яких також значно варіюється. Це такі способи:

* словоскладання (одно та багатокомпонентне) – 230 одиниць (76% вибірки);
* лексико-семантична транспозиція(вторинна номінація) – 12 одиниць (4% вибірки);
* абревіація – 3 одиниці (1% вибірки);
* суфіксація – 34 одиниці (11 % вибірки);
* префіксація – 23 одиниці (7 % вибірки);
* суфіксація + префіксація – 5 одиниць(1,6 % вибірки). Проаналізуємо специфіку використання зазначених способів.

«Словоскладання є способом словотвору, за якого відбувається поєднання двох або більше базових морфем» [4, c. 21]. Кожна з основ, які беруть участь у процесі словоскладання, може функціонувати самостійно, тобто утворювати незалежну лексичну одиницю. В сучасній англійській мові немає чітких правил оформлення складних слів: морфеми в їхньому складі можуть писатися не тільки разом або через дефіс, але й окремо. Отже, головним критерієм виділення складного слова є його семантична цілісність, під якою розуміємо номінацію за принципом «одна одиниця – одне поняття». Нагадаємо, що цей принцип відповідає найважливішій ознаці терміну – понятійності.

Деякі дослідники пропонують розрізняти складні слова та складні словосполучення за низкою ознак (фонетичною, орфографічною, морфологічною тощо) [9], але ми вважаємо, що семантична ознака є в даному випадку більш значущою та «перебиває» решту аргументів. Отже, у нашій вибірці представлені складні слова різних моделей:

*Двохкомпонентні****:***

Модель *N + N = N*: *assignment statements, Action Script, control flow, character string, command interpreter, country code, сable мodem, card reader, сircuit switching, coaxial cable, code page, common ground*.

Модель *Adj. + N = N: active matrix Algoritmic Language, arithmetic expression, coaxial cable, common ground, free software, fuzzy theory, generic protocols, hard drive, hot keys*.

Модель *N + Part. I = N: bus mastering, сircuit switching, Knowledge Engineering, packet switching.*

Модель *Part. II + N = N*:*, integrated circuit, leased line, managed code*. *Двохкомпонентні з прийменником:*

Модель *N + per + N = N: line per inch, kilobits per second, Megabits per second.*

Модель *N + in + N = N: boggedown in details*. *Трьохкомпонентні:*

Модель *Adj. + N + N = N: binary application Interface, Local Area Network, Literal Result Element, Media Access Control, Metropolitan Area Network, Mathematical Markup Language, Media Control Interface, Mobile Document Access, data-based knowledge, Multiple-Document Interface, Mobile Internet Device, multi- homed computer, Optical Character Recognition*.

Модель *N + N + N =N: Mail Delivery Agent, Model Driven Architecture, Module Interconnection Language, Network Address Translation, Network File System, Network Interface Card, radio access network, resource type element, data-based knowledge, document file icon*.

Модель *Adj. + Adj. + N = N: Liquid Crystal Display, Light-Emitting Diode*. Модель *Part.I + N + N = N: Manufacturing Enterprise Solutions, Manufacturing*

*Execution Systems.*

Модель *Part. II + N + N = N: parsed character data*. *Трьохкомпонентні з прийменником у:*

Модель *N + N + in + N = N: Knowledge Discovery in Databases*. Чотирьохкомпонентні:

Модель *N + N + N + N = N: Advanced Research Projects Agency*.

Решта термінологічних одиниць вибірки з програмування представлені окремими моделями, отже ми вважаємо за можливе просто навести декілька прикладів таких одиниць, не розділяючи їх за моделями утворення: *back-end database, data-based knowledge****,*** *dial-up connection, just-in-time compiler, just-in- time printing, Knowledge Based-System, off-line, public-domain software*.

Абревіація та акронімія є двома різновидами наймолодшого в діахронічному аспект способу словотвору, який полягає в поєднанні початкових літер (однієї або, рідше, кількох) декількох слів, що виступають засобом номінації одного об’єкту.

У сучасних термінологіях абревіація та акронімія виступають доволі продуктивним способом словотвору (що підтверджують, зокрема, дані і нашої вибірки). Це пояснюється передусім складністю понять сучасної науки та техніки, для вираження яких необхідні багатолексемні сполучення, отже, абревіація та акронімія виступають своєрідним засобом реалізації закону економії мовленнєвих зусиль.

За думкою А. Мартіне, рушійною силою мовних змін виступає «постійне протиріччя між потребами спілкування людини та її намагання звести до мінімуму свої розумові та фізичні зусилля. Внаслідок цього в процесі вживання мови мовець здійснює відбір найбільш раціональних для цілей спілкування засобів» [15, с. 533].

Наведемо декілька прикладів термінологічних абревіатур та акронімів нашої вибірки:

MS *– mili second; GB – giga byte; PB – peta byte.*

Продуктивним способом утворення термінів нашої вибірки є лексико- семантична транспозиція (або вторинна номінація), під якою розуміємо

«використання в акті номінації фонетичної форми вже існуючої одиниці і якості імені для нового об’єкта позначення» [15, с. 337].

Наведемо приклад: *hot keys* – гарячі клавіші (тобто більш задіяні клавіші). В звичайній мові, не програмування, *hot* – гарячий (вимір температури), *keys* – ключі (для замків).

Можливість утворення вторинних номінативних одиниць зумовлена на семіотичному рівні асиметричністю мовного знака (тобто відсутністю повної відповідності між планом вираження – формою та планом змісту – значенням), а на когнітивному рівні здатністю мозку людини до асоціацій. Унікальні здібності мозку та систематизація понять у пам’яті багато в чому визначаються асоціаціями. Асоціації становлять собою «мільярди зв’язків або поєднань, які сприяють організації більш менш стійких образів пам’яті з різними порціями інформації» [15, с. 48–52].

Таким чином, сама індивідуальна пам’ять людини є сховищем асоціацій. Асоціація як психолінгвістичний феномен становить зумовлений попереднім досвідом зв’язок уявлень, завдяки якому одне уявлення, з’являючись у свідомості індивіда, викликає інше. Формуючи висловлення, мовець розраховує на те, що його асоціації стосовно певного об’єкта матимуть збіг з асоціаціями інших учасників комунікації, що є запорукою інтерпретації подібних новотворів. У програмуванні виникнення нових значень термінів обумовлене потребою номінації нових понять і явищ, що виникають за умов об’єктивної дійсності чи необхідності розвитку синонімічних засобів певної мови, внаслідок чого утворюється і починає функціонувати номінативно-когнітивна метафора. Даний різновид метафори створюється на основі асоціативно-образного мислення, за допомогою 1) перенесення ; 2) звуження/ спеціалізація; 3)розширення/генерація. На перших етапах свого функціонування вона усвідомлюється носіями мови як семантично двопланове утворення. Надалі у процесі вживання номінативно- когнітивна метафора «звільняється від семантичної подвійності. Функція номінації, спрямована на відображення світу як об’єктивного даного, не сполучається із суб’єктивно орієнтованим сприйняттям дійсності» [17, с. 83]. У

зв’язку з цим даному виду метафори властива експресивна нейтральність, що відповідає принципу стилістичної нейтральності терміну.

Когнітивний аспект вторинної термінологічної номінації полягає у тому, що вона надає можливість проникнути у глибини людської свідомості, збагатити наші відомості про рівень розвитку науки в момент створення слова, а також уявлення про об’єкт в період його номінації. Для термінів-метафор характерна чітка демонстрація належності до понятійної сфери тієї чи іншої галузі знань.

Вибір за основу порівняння ознаки денотату, номінованої прямим значенням відповідного загальновживаного слова, передбачає осмислення характеру зв’язку між предметами чи явищами позамовної дійсності. Яскравість, образність вербальної ознаки, обраної мовцем, який у цьому випадку перетворюється на суб’єкта номінації, складає основу семантичного механізму формування термінологічного значення. При цьому, як показують наші спостереження, слово з порівняно простою семантичною структурою має більше шансів стати терміном.

Наведемо декілька прикладів таких новотворів:

*Debugger* – комп’ютерна програма що створена для позбавлення вірусів. У звичайній мові, *bug* – жук (вид комахи). Префікс *De* – означає позбавлення. Тож, ми можемо бачити аналогію між мовою звичайною та програмування, і на нашу думку термін *Debugger* має зміст позбавлення програмного забезпечення від вірусів, так само як люди позбавляють від шкідників, у нашому випадку від жуків, корисні рослини чи будівлі.

Наступний термін є похідним від попереднього, а саме застосування програми.

*Debugging –is work of the Debugger programm;*

В нашій вибірці є кілька одиниць, структурний статус яких доволі важко визначити. Справа у тому, що ці одиниці, по-перше, не мають деривативного характеру (принаймні такого, що можна визначити не вдаючись до складного етимологічного аналізу), а, по-друге, функціонують в сучасній англійській мові

виключно як терміни. Проте, саме етимологічний характер дозволяє зрозуміти походження та структуру таких термінологічних одиниць, наприклад:

*Star Schema – is the simplest style of data mart schema.*

Таким чином, ми схильні віднести ці термінологічні одиниці до лексико- семантичних транспозицій, в яких переосмислення або відбулося на іншомовній основі (до запозичення) або досить давно, втративши, таким чином, свою прозорість.

* 1. **Семантичний аналіз одиниць вибірки**

Структурні та семантичні характеристики лексичної одиниці (план вираження та план змісту) існують у тісному зв’язку – принаймні для похідних та складних слів, якими й є одиниці нашої вибірки. Структурний аналіз у багатьох випадках неможливий без урахування семантичної складової аналізованих слів (як ми мали змогу побачити у попередньому підрозділі), так само і семантичний аналіз полегшується за рахунок прозорості морфологічного членування лексем. Отже, виходячи з таких міркувань, логічним наступним кроком вважаємо проведення семантичного аналізу термінів у галузі програмування.

У перспективі перекладу семантичний аналіз є необхідним задля максимально повного відтворення значення вихідної одиниці у її цільовому відповіднику. У цьому контексті пригадуються положення семантичної теорії перекладу, згідно з якою «припускається, що процес перекладу може здійснюватися у два етапи. На першому етапі перекладач визначає семний склад відрізка оригіналу та вирішує, які із встановлених ним сем є релевантними, тобто суттєвими або важливими для здійснення міжмовного спілкування, а отже, мають обов’язково бути відтворені у перекладі. На другому етапі у МП підбираються одиниці, до значення яких входить максимальна кількість релевантних сем оригіналу. Тобто ступінь близькості перекладу до оригіналу (ступінь еквівалентності) визначається кількістю спільних сем» [13, с. 123].

Існують різні підходи до проведення семантичного аналізу. Меті та завданням нашої роботи найкраще підходить аналіз за номінативним критерієм, який передбачає виокремлення головних лексико-семантичних груп (далі ЛСГ) термінологічних одиниць.

Проаналізуємо загальні принципи такого аналізу. Його основою виступає розподіл аналізованих одиниць на лексико-семантичні класи як носії найбільш загальних семантичних ознак. Наступним етапом є класифікація одиниць різних частин мови за тематичними групами на основі релевантних ним семантичних категорій та мовних опозицій. «Нарешті, завершальним етапом є виділення в межах тематичних груп ЛСГ, що протиставлені одна одній за диференційним (мотивуючим) критерієм» [13, c. 97].

За морфологічним критерієм більшість одиниць нашої вибірки є іменниками, що дозволяє нам уникнути суперечки стосовно морфологічної приналежності терміну як об’єкта дослідження. Нагадаємо суть справи. Серед фахівців викликає питання стосовно того, якими частинами мови можуть бути терміни. Ряд науковців [11] вважають, що термінами є лише іменники, оскільки вони мають номінативний характер. Інші припускають, що термінами можуть бути і слова, що належать до різних частин мови, насамперед, до повнозначних. Так, на думку О. С. Ахманової, прикметники та дієприкметники також можуть використовуватись у функції терміна [2].

Як стверджував Л.В. Щерба, усім мовам властиві предметність, дія, якість [22]. Д.С. Лотте виділяв чотири основні категорії – категорію предметів, категорію процесів (явищ), категорію якостей, категорію величин, що, на його думку, найбільше чітко виявляються в будь-якій терміносистемі [14]. Визнаючи зв’язок між системою субмови науки і мовною системою в цілому, у даний час у термінознавчих дослідженнях затвердилася думка про те, що термінами можуть бути слова, представлені різними частинами мови, знаменними насамперед, якщо в складі визначеної термінології вони набули термінологічного змісту і служать для вираження системних відносин (дія – ознака дії – агент дії тощо).

Іменники нашої вибірки представлені наступними трьома тематичними групами:

1. найменування абстрактного поняття;
2. найменування особи;
3. найменування предмету.

Пропонуємо наступну номенклатуру ЛСГ в межах цих тематичних груп.

1. Найчисельніша тематична група «найменування абстрактного поняття» представлена наступними лексико-семантичними групами:

а) програма / операція:

*Active matrix* – активна матриця

*Application binary Interface* – двійковий інтерфейс застосувань

*Assignment statements* – оператори присвоєння *Add-in program* – програмні розширення *Aperture lattice* – апертурна решітка

*Back-end database* – прикладна частина бази даних *Back end software* – кiнцеве програмне забезпечення *Back-end systems* – виконуючі системи

*Bad sector* – дефектний сектор *Bar code* – штриховий код *Compiler* – компiлятор

*Character string* – строка символiв

*Chunker* – програма-фрагментатор

*Command line interface* – інтерфейс командного рядка

*Configuration file* – файл конфігурації *Dashboard* – інструментальна панель *Database* – база даних

*Debugger* – налагоджував

*Dial-up connection* – комутоване з’єднання по телефонній лінії

*Document file icon* – піктограма файла документа

*Free software* – вільне програмне забезпечення

*Freeware* – безкоштовне програмне забезпечення

*Malware –* шкідливе програмне забезпечення б) виміри/одиниці обчислення:

*Arrays* – масив

*Bit-map* – бітовий массив *Constant* – константа *Decibel* – децибел *Domain* – домен

*Element* – елемент *Frame* – кадр *Gbyte* – гігабайт

*GigaHertz* – гігагерц

*Hertz* – герц *Kilobit* – кілобіт *Kilobyte* – кілобайт

*Kilohertz* – кілогерц

*Micro* – мікро

*Ms* – мілісекунда *Nanosecond* – наносекунда *PByte* – петабайт

в) процес / система:

*Algorithmic Language* – алгоритмічна мова *Augment* – збільшувати, збільшення *Action Script* – скриптова мова

*Backup* – резервна копія

*Binary digit* – двійковий знак, двійкова цифра, двійковий розряд

*Browser* – браузер

*Bundled software* – стандартне програмне забезпечення

*Bytecode* – байткод

*Bus mastering* – одноосібне керування шиною

*Computation* – обчислення

*Command interpreter* – командний інтерпретатор, командний процесор

*Cycle –* цикл

*Circuit switching* – комутація каналів *Concept search* – концептуальний пошук *Debugging* – налагодження

*Download* – завантаження

*Extrapolate* – екстраполювати

Escape sequence – Послідовність символів

*Environment* – середовище Obstruction – перешкода, бар’єр

1. Друга за кількістю тематична група «найменування предмета» представлена наступними ЛСГ:

а) інструмент/технічна складова/деталь:

*Chip* – чіп

*Chipset* – набір мікросхем

*Coaxial cable* – коаксіальний кабель

*Fiber* – оптичне волокно

*Hard drive* – накопичувач на жорсткому диску

*Liquid Crystal Display* – дисплей, рідкокристалічний дисплей *Light-Emitting Diode* – світлодіодний індикатор *Microprocessor* – мікропроцесор

*Motherboard* – материнська плата *Multi-touch screen* – сенсорний екран *Northbridge* – північний міст

б) пристрій:

*Cable мodem* – кабельний модем *Card reader* – кардрідер *Communicator* – комунікатор *Hаrd disk* – жорсткий диск

*Hot keys* – гарячі клавіші

*Laptop* – лептоп

*Line Printer* – порядковий принтер

*Microcomputer* – мікрокомп’ютер

*Mobile computer* – кишеньковий комп’ютер *Mobile device* – мобільний пристрій *Netbook* – нетбук

*Notebook* – ноутбук

1. Тематична група особи ми виділяємо наступною ЛСГ а) фахівець/майстер:

*Knowledge engineer* – інженер зі знань, когнітолог

*Postmaster* – фахівець з питань керування поштовими сервісами

*Stakeholder* – організатор.

**Висновки до Розділу 2**

1. Вивчення структурних характеристик термінів може бути корисним з точки зору з’ясування: 1) продуктивних / малопродуктивних / непродуктивних способів / засобів номінації термінотворення; 2) ступеню складності концептуального апарату та понять даної галузі (на основі сполучення структурного аналізу з семантичним); 3) оптимальних стратегій та способів перекладу для створення / підбору відповідника з максимальним рівнем адекватності.
2. Для утворення термінологічних одиниць у галузі програмування були застосовані такі способи: словоскладання (одно та багатокомпонентне) – 230 одиниць (76% вибірки); лексико-семантична транспозиція(вторинна номінація) – 12 одиниць (4% вибірки); абревіація – 3 одиниці (1% вибірки); суфіксація – 34 одиниці (11 % вибірки); префіксація – 23 одиниці (7 % вибірки);суфіксація + префіксація – 5 одиниць(1,6 % вибірки).
3. Згідно з положеннями семантичної теорії перекладу, семантичний аналіз вихідної одиниці є необхідним для того, аби підібрати / створити у цільовій мові відповідник із максимально подібним набором елементарних змістів (сем). За рахунок цього забезпечується адекватність перекладу.
4. Терміни-іменники нашої вибірки представлені наступними трьома тематичними групами: (1) найменування абстрактного поняття;

(2) найменування особи; (3) найменування предмету. Група «найменування абстрактного поняття» включає такі ЛСГ: програма / операція; виміри/одиниці обчислення; процес / система. Група «найменування предмету» представлена такими ЛСГ: інструмент/технічна складова/деталь; пристрій/девайс. Група

«найменування особи» представлена ЛСГ «фахівець/майстер».

**РОЗДІЛ 3**

**АНАЛІЗ ПЕРЕКЛАДУ ТЕРМІНІВ У ГАЛУЗІ ПРОГРАМУВАННЯ**

* 1. **Способи перекладу термінів зі словника у галузі програмування**

Комп’ютерна техніка і технологія міцно увійшла в наше життя. Але розвиток науково технічного прогресу не був би можливим без автоматизації обчислювальних процесів. Саме потреба в автоматизації обчислювальних процесів стала початковим імпульсом у розвитку програмування (комп’ютерів та електронно-обчислювальних машин). Далі з’явилася думка про автоматизацію процесів управління механізмами і машинами на основі комп’ютерних технологій [29, c. 171]. Після отримання перших результатів та реалізації задумів компаній що працюють у галузі програмування, зробили наступний крок – створили комп’ютери та комп’ютерну техніку для задоволення найрізноманітніших потреб людей: ігри, відеофільми, музика тощо [38, c. 73].

Оскільки ринок промислових комп’ютерних розробок не здатній дати достатньо коштів для інтенсивного розвитку програмування та кібернетики, комп’ютерні компанії зробили ставку на індустрію комп’ютерних розваг. На базі комп’ютерного програмування з’явилися найрізноманітніші розваги для дітей юнаків, та дорослих похилого віку.

Комп’ютерну лексику починають використовувати не лише програмісти, але й люди, які зовсім не мають ніякого відношення до комп’ютера, оскільки це явище досить швидко вливається в життя сучасного суспільства. Однак треба усвідомлювати що виникають певні труднощі при перекладі лексики у галузі програмування, деякі з них ми вже розгледіли в розділу 1. А зараз ми звернемо увагу на те як перекладачі перекладають таку лексику безпосередньо у галузі програмування.

Перекладачеві інколи доводиться самостійно створювати еквівалентні терміни рідною мовою, причому основна проблема, яку доводиться розв’язувати перекладачеві науково-технічної літератури, полягає саме у перекладі термінів,

що веде до появи неологізмів, які становлять досить великий відсоток всієї лексики у перекладах, перекладачі змушені винаходити засоби передачі відсутнього терміна для подальшої його стандартизації та закріплення у відповідних термінологічних словниках. Процес набуття нової лексики, більша частина якої є переробленими на місцевий лад англійськими словами, породило спочатку запозичення, а потім і терміни нової субкультури. І зрозумілі більшості слова, такі, як «юзер», «вінди», користуються більшою популярністю, ніж їхні літературні еквіваленти «користувач», «операційна система *Windows*» [5]. Комп’ютерна термінологія повинна стати об’єктом уваги вчених – мовознавців, перекладознавців, оскільки, як видно з інших галузей науки, спеціальна лексика іноді проникає в літературну мову і закріплюється в ній на довгі роки [34, c. 50]. По-перше слід розгледіти сам термін "програмування" адже адекватний переклад неможливим без додаткових знань, пов’язаних з походженням, класифікацією, функціонуванням та особливостями перекладу саме термінів.

Програмування (*programming*) – діяльність, метою якої є опис процесів обробки даних (у широкому значенні) [39, c. 144]. Всі технічні операції, необхідні для створення програми, включаючи аналіз вимог і всі стадії розробки й реалізації: проектування, написання, налагодження, тестування, документування й супровід ПЗ (у вузькому значенні) [26, c. 60]. Кодування й тестування програми в рамках деякого конкретного проекту. Основними проблемами, що виникають при програмуванні на сучасному рівні, є: а) створення специфікацій; б) облік і використання особливостей вибраної мови програмування; в) проблеми іменування; г) розробка інтерфейсів; д) облік взаємодії протоколів і стандартів; е) вибір необхідної архітектури застосування, яка визначається архітектурою використовуваної застосуванням апаратної системи. Програмування (у «малому») [37 c. 210]. Для нього характерні такі ознаки: а) програмний код розробляється єдиним програмістом або невеликою групою. Окремий індивідуум може зрозуміти всі аспекти проекту від початку до кінця; б) основна проблема при розробці полягає в проектуванні програми і написанні коду алгоритмів для вирішення поставленої задачі [41, c. 32].

Програмування (у «великому»). Наділяє проект такими властивостями: а) програмна система розробляється великою командою програмістів. При цьому одна група може займатися проектуванням системи (або написанням специфікацій до системи), інша – здійснювати написання кодів окремих компонентів [40, c. 24]. А третя – об’єднувати фрагменти в кінцевий проект; б) немає жодної людини, яка б знала все про виконуваний проект. Основна проблема в процесі розробки такого ПЗ – керування проектом і обмін інформацією між групами й усередині груп: програмування веб-застосувань (веб-застосування); програмування візуальне (візуальне програмування); програмування декларативне (декларативні мови); програмування екстремальне (екстремальне програмування); програмування імперативне (процедурні мови);програмування компонентно-орієнтоване (компонентно-орієнтоване програмування); програмування конкурентне (конкурентне програмування); програмування лінійне (лінійне програмування); програмування логічне (декларативні мови, логічне програмування); програмування математичне (математичне програмування); програмування методологія (методологія програмування); програмування модульне (модульне програмування); програмування наукове (наукове програмування); програмування об’єктно- орієнтоване (об’єктно-орієнтоване програмування); програмування парадигми (парадигми програмування); програмування паралельне (паралельне програмування); програмування процедурне (процедурні мови); програмування розподілене (розподілене програмування); програмування сервіс-орієнтоване (сервіс-орієнтований веб); програмування системне (системне програмування); програмування структурне (структурне програмування); програмування функціональне (функціональна мова програмування); продуктивність (системи) (*performance*) [44, c. 57].

Об’єм робіт, що виконуються комп’ютером за певний час. Звичайно вимірюється в одиницях виконуваних ним операцій за секунду. Істотно залежить від типу вирішуваних задач. Продуктивність комп’ютера при виконанні цілочисельних обчислень значно вища в порівнянні з продуктивністю при

розв’язанні завдань, що вимагають обробки чисел з плаваючою точкою (рухомою комою) [48, c. 199].

Показник того, наскільки швидко працює система, тобто наскільки швидко вона виконує покладені на неї завдання. Продуктивність вимірюється по безлічі різних показників, що враховують вплив цілого ряду чинників, таких, як робоче навантаження, апаратна конфігурація (наприклад, тактова частота процесора, швидкість системної шини, швидкість передачі даних в мережі), а також кількістю звернень і відповідно операцій над БД [28, c. 121]. Але те, як саме впливають на продуктивність системи ці чинники, залежить від якості реалізованої архітектури рішення (застосування), а також вимог, що висувають до рішення на етапі проектування.

При дослідженні комп’ютерної термінологічної системи слід також звернути увагу на наступну низку особливостей:

1. Процес аналізу комп’ютерної лексики ускладнюється різноманітністю і неоднозначністю термінології (професіоналізми, жаргонізми, арготизми, сленгізми).
2. Комп’ютерна термінологія у своєму функціонуванні і словотворі підпорядкована законам української мови. Зокрема, найпоширенішими способами словотворення у комп’ютерній лексиці є афіксальний, безафіксний, а також лексико-семантичний.
3. Для комп’ютерної лексики характерне застосування мовної гри і засобів мовної експресії.
4. Причини запозичення з англійської мови зумовлені розвитком комп’ютерних технологій в англомовних країнах, а саме – у США.

Для проведення нашого дослідження словника ми відібрали терміни, та поставили перед собою завдання, переклад цих термінів. При перекладі одиниць вибірки застосовані такі способи перекладу та трансформації лексичного рівню: Калькування + транскодування (50 одиниць або 16,6%); калькування (30 одиниць або 10%); транскодування (60 одиниць або 20%); вживання еквіваленту з мови перекладу (155 одиниць або 51,6%); описовий переклад (3 одиниці або 1%). Ми

визначили що найбільш вживаним засобом є вживання еквіваленту з мови перекладу.

Еквівалентний переклад – значення англійського слова повністю відповідає значенню українського слова. Цей вид перекладу досить поширений при перекладі англійських комп’ютерних термінів, хоча в українській мові існує не надто велика кількість слів-відповідників у цій галузі. Ми можемо навести наступні приклади еквівалентного перекладу:

*keyboard* – клавіатура; *memory* – пам’ять; *message* – повідомлення; *mode* – режим;

*mouse* – маніпулятор; *notepad* – записна книжка; *pitch* – рівень, висота; *screen* – екран;

*desktop –* робочий стіл;

*network* – мережа; *bar* – панель; *drive* – дисковод; *error* – помилка; *lock* – блокування.

Отже, у нашому дослідженні ми опрацювали лише невелику частку того багатого матеріалу, яким є для вчених мовознавців комп’ютерна термінологія, але можемо зробити кілька висновків. Із прикладів, які ми опрацювали, можна вивести таку закономірність: найбільш уживаним шляхом перекладу є транскодування, менш вживаним – калькування. Комп’ютерна термінологія є, мабуть, найдинамічнішою із термінологічних систем, і в ній навряд чи колись можна буде поставити крапку та вважати дослідженою, оскільки інновації в комп’ютерній техніці розвиваються досить стрімко і не дають ні хвилини відпочити лінгвістам, зокрема перекладачам з англійської мови (звідки й

потрапляє до нас більшість комп’ютерних термінів), які покликані тлумачити ці терміни, перекладати їх, вникати подекуди не лише в мовознавчі питання (який спосіб перекладу використати), а й у технічні деталі описуваного поняття. Другим по кількості є транскодування.

Транскодування, коли звукова та/або графічна форма слова вихідної мови передається засобами абетки мови перекладу. Транскодування неологізмів відбувається в перекладі в тих випадках, коли в культурі і, зокрема, науці країни мови перекладу відсутнє відповідне поняття і відповідний перекладний еквівалент, а перекладач не може підібрати слово або слова в мові перекладу, які б адекватно передавали зміст поняття і відповідали вимогам до термінотворення. При транскодуванні літерами мови може передаватися вся форма, або більша її частина. Іноді застосовується змішане транскодування, коли більша частина транскодованого слова відбиває його звучання у вихідній мові, але разом з тим передаються й деякі елементи його графічної форми Особливо часто транскодування термінів відбувається в тих випадках, коли термін у мові перекладу складається з міжнародних терміноелементів латинського або давньогрецького походження. Транскодування поділяється на чотири підвиди. Розглянемо кожен із них окремо:

Транслітерація: *commutator* – комутатор; *processor* – процесор; *operator* – оператор; *monitor* – монітор; *printer* – принтер; *indicator* – індикатор; *buffer* – буфер; *portal* – портал; *server* – сервер; *laser* – лазер; *multimedia* – мультимедія; *decoder* – декодер; *port* – порт; *plotter* – плотер; *scanner* – сканер.

Необхідно зазначити деякі особливості у перекладі цих слів-термінів, а саме: 1) подвоєння приголосних між голосними не передається, як, наприклад, у слові «комутатор»; 2) літера *r* у кінці зазвичай передається, незалежно від того, чи вимовляється вона у слові-джерелі, наприклад, монітор.

Транскрибування: Цей прийом становить рідкість в практиці технічного перекладу. При перекладі таким способом, як транслітерація, не слід забувати про викривлення змісту, наприклад, таких як *contribution, data, instance, stimulation* тощо.

При транскодуванні літерами мови може передаватися вся форма, або більша її частина. Іноді застосовується змішане транскодування, коли більша частина транскодованого слова відбиває його звучання у вихідній мові, але разом з тим передаються й деякі елементи його графічної форми .

Транскодування термінів відбувається при перекладі в тих випадках, коли в культурі і, зокрема, науці країни мови перекладу відсутнє відповідне поняття та відповідний перекладний еквівалент, а перекладач не може підібрати слово або слова в мові перекладу, які б адекватно передавали зміст поняття і задовольняли вимогам до термінотворення. Оскільки при транскодуванні слово має одне значення, то такий спосіб перекладу доцільно застосовувати, коли в мові перекладу необхідно створити чітко однозначний термін, наведемо приклади з нашого дослідження: *laptop* – лептоп; *browser* – браузер; *site* – сайт; *provider* – провайдер; *cluster* – кластер; *display* – дисплей; *user* – юзер; driver – драйвер; *cartridge* – картридж; *computer* – комп’ютер.

Цьому шляху перекладу властива така особливість у перекладі українською мовою, як передача в будь якому випадку літери, наприклад, *driver*

* драйвер.

Змішане транскодування: *on-line* – он-лайн; *interface* – інтерфейс; *organizer*

* органайзер; *device* – девайс;*chat* – чат; *chipset* – чіпсет; *adaptor* – адаптор; *chorus*
* хорус.

Адаптоване транскодування: *profile* – профіль; *matrix* – матриця; *domain* – домен; *command* – команда; *card* – карта; *menu* – меню; *viewer* – в’юер.

Цьому шляху перекладу притаманні наступні особливості:

1) використання в українській мові пом’якшення в кінці слова, яке відсутнє в англійському слові, наприклад, модуль; 2) наявність родового закінчення в мові перекладу, наприклад, модифікація; 3) подвоєння приголосних між голосними не передається в українській мові. Наступний за кількістю вживаності засіб у нашій вибірці є калькування + транскодування. Менший по кількості способом вибірки є калькування.

Калькування – переклад лексичних одиниць оригіналу шляхом заміни їх складових частин – морфем або слів – їх лексичними відповідниками в мові перекладу. Калькування як прийом перекладу частіше застосовується в перекладі складних слів (термінів). Воно може застосовуватися також стосовно тільки одного з компонентів складеного слова (терміна). Досить часто калькування застосовуються в перекладі до тих складних термінів, що утворені за допомогою поширених загальнонародних слів. У ряді випадків використання прийому калькування супроводжується зміною послідовності калькованих елементів. Калькування можна застосовувати тільки тоді, коли утворений таким чином перекладний відповідник не порушує норми вживання і сполучуваності слів в українській мові. Обираючи між транскодуванням та калькуванням, перевагу віддають калькуванню, так як в результаті транскодувань нерідко утворюються одиниці, що не мають сенсу у мові перекладу, свого роду псевдослова. Калькування не завжди буває звичайною механічною операцією з метою перенесення вихідної форми у мову перекладу. Дуже часто доводиться вдаватись до різного роду трансформацій. В першу чергу це стосується зміни відмінкових форм, кількості слів у словосполученні, афіксів, порядку слів, морфологічного або синтаксичного статусу слова. Нерідко в процесі перекладу транскрипція і калькування використовується одночасно. Наводимо наступні приклади: *control panel* – панель управління; *matrix printer* – матричний принтер; *hot keys* – гарячі кнопки; *file system* – файлова система; *data warehouse* – інформаційне сховище; *artificial neutral network* – штучна нейтронна мережа; *composite key* – композитний ключ; *computer network* – комп’ютерна мережа; *mailbox* – поштова скриня; *error checking* – контроль помилок; *ring network* – кільцева мережа; *network; neighborhood* – мережеве оточення.

Описовий переклад характеризується такими недоліками: 1) при його застосуванні можливо неточне або нечітке тлумачення змісту поняття, позначеного неологізмом, 2) описове словосполучення порушує таку вимогу до термінів, як стислість (а тому такі багатослівні терміни не мають дериваційного потенціалу, тобто, від них важко, якщо взагалі можливо, утворювати похідні

терміни. Якщо словник не дає точного еквівалента тому чи іншому терміну, або застосування всіх вищезазначених способів є недоречним, можливими є інші прийоми перекладу на лексико-семантичному та лексико-граматичному рівнях [5], до яких відносять конкретизацію та генералізацію.

* 1. **Способи перекладу термінів у тексті з галузі програмування**

У нашому дослідженні ми взяли на розгляд частину тексту з галузі програмування за для проведення аналізу по відповідності значення терміну та його перекладу. Раніше ми розглядали вибірку зі словника програмування, а саме у розділі 2 та 3.1, тож перевіримо переклад наданий в тексті та словнику.

З тексту ми обрали 17 термінів деякі з яких представлені також і в словнику.

1. *Element* – елементи – елемент
2. *boggeddown* in details – надмір деталей – (надмір деталей ) – словник 3 *variable* – змінна

4 *control flow* – керування потоком – (керування струмом) – словник 5 *input* – ввод – (ввід) – словник

1. *output* – вивод
2. *compile* – компiлювати
3. *run the program* – запустити программу
4. *function* – фунеція
5. *computation* – обчислення 11 *compiler* – компілятор
6. *arguments* – аргумент
7. *escape sequence –* Екранована послідовність – послідовність символів 14 *constant* – константа

15 *costant* – стала 16 *byte* – байт

*17 array* – масив

Розглянувши переклад цих термінів, можемо побачити що терміни *control flow, input, escape sequence*, мають розбіжний зі словником переклад. Тож, доречно припустити що переклад цих термінів є частково у стані розробки. Ми виокремили лише три з сімнадцяти обраних, але навіть така кількість є ознакою існування недоробок в перекладі і в тексті в цілому, як і в мові програмуванні взагалі є достатня кількість таких випадків.

Наступне, що ми розглянемо, це стабільність перекладу за контекстом. У нашій вибірці ми налічили один випадок, що вартий уваги. Термін *constant* перекладається, як «константа» та як «сталий». Тож, ми можемо бачити два різних способи перекладу та невизначеність автора єдиного еквівалента. Наведемо уривок тексту.

Англійський варіант.

*We want to get you as quickly as possible to the point where you can write useful programs, and to do that we have to concentrate on the basics: variables and* ***constants****, arithmetic, control flow, functions, and the rudiments of input and output.*

Український переклад

*Ми хочемо привести вас якнайшвидше до того пункту, де ви зможете самостійно писати корисні програми, i щоб досягти цього, ми повинні зосередитись на базових речах: змінних i* ***константах (сталих)****, арифметиці, керуванню потоком виконання, функціях i найпростіших операціях вводу та виводу.*

Таким чином, можемо бачити існування проблеми стабільності у мові програмування. Хоча проблема розробки є, по нашим підрахункам, суттєвіша, досконалість перекладу у цій галузі залежить як від розробки так і від стабільності перекладу.

**Висновки до розділу 3**

1. Термінологія програмування є постійним джерелом неологізмів, тож потребує пильного спостереження з боку науковців за для поліпшення перекладу.
2. У перекладі з галузі програмування є питання стосовно розробки та стабільності перекладу. Від швидкості вирішення цих питань залежить порозуміння між діячами галузі програмування.
3. За результатами нашого дослідження нами було виявлені наступні способи перекладу: Калькування + транскодування (50 одиниць або 16,6%); калькування (30 одиниць або 10%); транскодування (60 одиниць або 20%); вживання еквіваленту з мови перекладу (155 одиниць або 51,6%); описовий переклад (3 одиниці або 1%). Ми визначили що найбільш вживаним засобом є вживання еквіваленту з мови перекладу.. В нашому дослідженні ми вирішили що найуживанішим способом, у перекладі термінології у галузі програмування є вживання еквіваленту з мови перекладу (155 одиниць або 51,6%) а найменш вживаним є описовий переклад (3 одиниці або 1%).
4. У розділі 3.2. ми детально розглянули текст з галузі програмування та визначили що деякі з термінів мають не тільки різний переклад у тексті та словнику але і у самому перекладі термін може перекладатися по різному що свідчить про недосконалість та не відповідність визначеному нами поняття терміну як постійного відповідника.

**ЗАГАЛЬНІ ВИСНОВКИ**

Спираючись на дослідження науковців, що наведені у розділі 1, та керуючись власною думкою, ми визначили поняття терміну як слова або словосполучення, що найбільш влучно визначає сутність галузевих понять, та назв речей, а в аспекті перекладу характеризується наявністю постійного відповідника.

Проаналізувавши теоретичні джерела у галузі програмування та наведені приклади в галузі термінології програмування, нами було зроблено висновок, що першорядними труднощами у процесі перекладу досліджуваних одиниць є такі фактори як: наявність абревіатур та скорочень в даній галузі; потреби враховувати особливості перекладу складних граматичних конструкцій; адекватність вибору правильного значення полісемантичного слова; велика кількість безеквівалентних лексичних одиниць у складі комп’ютерної термінології англійської мови; відсутність єдиної дійсної моделі перекладу комп’ютерних текстів; морфологічна будова і семантичні особливості термінів; постійне оновлення лексики і поява комп’ютерних неологізмів. Окремим пунктом виступає обов’язкова індивідуальна грамотність перекладача, що є ключовим аспектом успішної комунікації.

Слід також зазначити, що абсолютно всі перераховані вище особливості перекладу англійської комп’ютерної лексики українською мовою мають надзвичайно вагоме значення у сфері міжкультурного спілкування. Правильний підхід до труднощів у процесі перекладу комп’ютерної лексики гарантує перекладачеві успішний результат у сприйманні та наданні необхідної інформації.

Вивчення структурних характеристик термінів може бути корисним з точки зору з’ясування: 1) продуктивних / малопродуктивних / непродуктивних способів / засобів номінації термінотворення; 2) ступеню складності концептуального апарату та понять даної галузі (на основі сполучення структурного аналізу з семантичним); 3) оптимальних стратегій та способів перекладу для створення / підбору відповідника з максимальним рівнем адекватності.

Для утворення термінологічних одиниць у програмування були застосовані такі способи: словоскладання (одно та багатокомпонентне) – 230 одиниць (76% вибірки); лексико-семантична транспозиція(вторинна номінація)

* 12 одиниць (4% вибірки); абревіація – 3 одиниці (1% вибірки); суфіксація – 34 одиниці (11 % вибірки); префіксація – 23 одиниці (7 % вибірки);суфіксація + префіксація – 5 одиниць(1,6 % вибірки).

Згідно з положеннями семантичної теорії перекладу, семантичний аналіз вихідної одиниці є необхідним для того, аби підібрати / створити у цільовій мові відповідник із максимально подібним набором елементарних змістів (сем). За рахунок цього забезпечується адекватність перекладу.

Терміни-іменники нашої вибірки представлені наступними трьома тематичними групами: (1) найменування абстрактного поняття;

(2) найменування особи; (3) найменування предмету. Група «найменування абстрактного поняття» включає такі ЛСГ: програма / операція; виміри/одиниці обчислення; процес / система. Група «найменування предмету» представлена такими ЛСГ: інструмент/технічна складова/деталь; пристрій/девайс. Група

«найменування особи» представлена ЛСГ «фахівець/майстер».

Термінологія програмування є постійним джерелом неологізмів, тож потребує пильного спостереження з боку науковців за для поліпшення перекладу.

У перекладі з галузі програмування є питання стосовно розробки та стабільності перекладу. Від швидкості вирішення цих питань залежить порозуміння між діячами галузі програмування.

В нашому дослідженні ми виявили наступні способи перекладу: калькування + транскодування, калькування, транскодування, вживання еквіваленту та описовий переклад. За результатами нашого дослідження ми вирішили що найуживанішим способом, у перекладі термінології у галузі програмування є вживання еквіваленту з мови перекладу (155 одиниць або 51,6%) а найменш вживаним є описовий переклад (3 одиниці або 1%).

Детально розглянувши текст з галузі програмування у розділі 3.2. ми визначили що деякі з термінів мають не тільки різний переклад у тексті та словнику але і у самому перекладі термін може перекладатися по різному що свідчить про недосконалість перекладу та не відповідність визначеному нами понятті терміну як постійного відповідника.

**СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Авербух К. Я. Общая теория термина / К. Я. Авербух. – М. : МГОУ, 2006. – 252 с.
2. Ахманова О. С. Словарь лингвистических терминов / О. С. Ахманова. – М. : Сов. Энциклопедия, 1966. – 607 с.
3. Бесекирска Л. К вопросу об определении термина / Л. Бесекирска // Терминоведение. – 1996. – № 13. – C. 34–37.
4. Бушев А. Б. Лингводидактика перевода военного термина / А. Б. Бушев // Вестник военного университета. – М. : ВУ МО РФ, 2009. – № 3 (19). – С. 123–129.
5. Винокур Г. О. О некоторых явлениях словообразования в русской технической терминологии / Г. О. Винокур // Труды МН ФПИ. – М. : МН ФПИ, 1939. – Т. 5. – С. 3–54.
6. Вільна енциклопедія «Вікіпедія» [Електронний ресурс]. – Режим доступу : [http://uk.wikipedia.org.](http://uk.wikipedia.org/)
7. [Гуманитарная энциклопедия](http://gtmarket.ru/encyclopedia/) [Электронный ресурс] // Центр гуманитарных технологий, 2010–2017 (последняя редакция: 07.12.2017). Режим доступа : [http://gtmarket.ru/concepts/7192.](http://gtmarket.ru/concepts/7192)
8. Карабан В. І. Переклад англійської наукової і технічної літератури : [у 2-х ч.] / В. І. Карабан. – Вінниця : Нова книга, 2001. – Ч. 2. Лексичні, термінологічні та жанрово-стилістичні труднощі. – 302 с.
9. Квитко И. С. Термин в научном документе / И. С. Квитко. – Львов : Вища школа, Издательство при Львовском университете, 1976. – 127 с.
10. Комиссаров В. Н. Теория перевода. Лингвистические аспекты / В. Н. Комиссаров. – М. : Высшая школа, 1990. – 252 с.
11. Кулебакин В. С. Работы по построению научно-технической терминологии в СССР и советская терминологическая школа / В. С. Кулебакин, А. Я. Климовицкий // Лингвистические проблемы научно-технической терминологии. – М. : Наука, 1970. – С. 15–19.
12. Лантюхова Н. Н. Термин: Определение понятия и его сущностные признаки [Электронный ресурс] / Н. Н. Лантюхова, О. В. Загоровская, Т. А. Литвинова

// Вестник Воронежского института ГПС МЧС России. – 2013. – № 1 (6) – С. 41–44. – Режим доступа : <http://cyberleninka.ru/article/n/termin-opredelenie-> ponyatiya-i-egosuschnostnye-priznaki (дата обращения: 27.07.2016).

1. Лейчик В. М. Термины и терминосистемы – пограничная область между искусственным и естественным в языке / В. М. Лейчик // Вопросы терминологии и терминологической статистики. – Воронеж : Воронежский ун-т, 1976. – С. 3–11.
2. Лотте Д. С. Основы построения научно-технической терминологии / Д. С. Лотте. – М. : АН СССР, 1961. – 158 с.
3. Мартине А. Основы общей лингвистики / А. Мартине // Новое в лингвистике. – М. : Изд-во иностр. лит., 1963. – Вып. 3. – С. 366–566.
4. Ребрій О. В. Оказіоналізми в сучасній англійській мові (структурно- функціональний аналіз): автореф. дис. на здобуття наук. ступеня канд. філол. наук : спец. 10.02.04 «Германські мови» / О. В. Ребрій. – Харків, 1997.

– 18 с.

1. Ребрій О. В. Сучасні концепції творчості у перекладі : [монографія] / О. В. Ребрій. – Харків : ХНУ імені В. Н. Каразіна, 2012. – 376 c.
2. Реформатский А. А. Мысли о терминологии / А. А. Реформатский // Современные проблемы русской терминологии. – М. : Наука, 1968. – С. 163– 198.
3. Реформатский А. А. Что такое термин и терминология / А. А. Реформатский

// Вопросы терминологии. – М. : АН СССР, 1961. – С. 46–55.

1. Федоров А. В. Основы общей теории перевода / А. В. Федоров. – М. : Высшая школа, 1983. – 303 с.
2. Шелов С. Д. Терминология, профессиональная лексика и профессионализмы / С. Д. Шелов // Вопросы языкознания, 1984. – №5. – С. 76–87.
3. Щерба Л. В. Языковая система и речевая деятельность / Л. В. Щерба. – Л. : Наука, 1974. – 428 с.
4. Юшманов Н. В. Элементы международной терминологии : словарь- справочник / Н. В. Юшманов. – М. : Наука, 1968. – 72 с.
5. Barron D. W. Pascal – The Language and its Implementation [Electronic resource] / D. W. Barron. – 1981. – 301 p. – Access : https://books.google.com.ua/books/about/PASCAL.html?id=X6hQAAAAMAA J&redir\_esc=y.
6. Bell R. T. Translation and Translating: Theory and Practice / R. T. Bell. – L. : Longman, 2000. – 266 c.
7. Brian W. Why Pascal is Not My Favorite Programming Language / W. Brian // Archived 2009-04-28 at the Wayback Machine. – 1981.
8. Catford J. C. A Linguistic Theory of Translation / J. C. Catford – L. : Longman, 2001. – 266 c.
9. Dictionary of computer science, engineering, and technology / [Ed. Ph. A. Laplante]. – CRC Press, 2001. – 560 p.
10. Elliott J. The Steve Jobs way: iLeadership for a new generation / Jay Elliott; William L. Simon. – Philadelphia : Vanguard, 2011. – 71 p.
11. Fantinuoli C. Specialized Corpora from the Web and Term Extraction for Simultanious Interpreters / C. Fantinuoli // XML International – 2013. – P. 173– 190.
12. Felber H. Terminology Manual / H. Felber // General Information Programme and UNISIST. International Information Centre for Terminology / [ed. by H. Felber].

– Paris : Unesco : Infoterm, 1984. – 426 p

1. Forsyth R. S. Pascal in Work and Play / R. S. Forsyth. – London ; New York : Chapman and Hall, 1982. – 360 p.
2. Fortran Variable Declarations. Compaq Fortran / [Retrieved 19 November 2014. The form CHARACTER\*(\*) is an obsolescent feature in Fortran 95]. – Texas, Huston, US : Compaq Computer Corporation, 1999.
3. Galperin B. Large Eddy Simulation of Complex Engineering and Geophysical Flows / B. Galperin. – London : Cambridge, 1993. – 573 p.
4. Gerber A. MOF to EMF and back again / A. Gerber // Cooperative Research Centre for Enterprise Distributed Systems (DSTC). – 2011. – P. 56–60.
5. Graham D. Foundation of software testing ISTQB Certification / Graham D. – L. : Tomson, 2005 – 235 p.
6. Grogono P. Programming in Pascal / P. Grogono. – [Revised Edition]. – Mass. : Addison-Wesley, 1980. – 420 p.
7. Hacker B. C. American Military Technology: The Life Story of a Technology / Barton C. Hacker. – Westport: Greenwood Press, 2006. – 205 p.
8. Haines L. H. Serial compilation and the 1401 FORTRAN compiler / L. H. Haines

// IBM Systems Journal ; [This article was reprinted, edited, in both editions of Lee, John A. N. (1967). Anatomy of a Compiler. Van Nostrand Reinhold] – 1965.

– №4 (1). – P. 73–80.

1. Hervey S. Thinking Spanish Translation: A Course in Translation Method: Spanish into English / S. Hervey, I. Higgins, L. M. Haywood. – L.; NY : Routledge, 2008. – 415 p.
2. Interactive terminology for Europe [Electronic resource] // Online translator. – Access : [http://iate.europa.eu.](http://iate.europa.eu/)
3. Internal Translator (IT) A Compiler for the IBM 650 [Electronic resource] / [by

A. J. Perlis, J. W. Smith, and H. R. Van Zoeren]. – Access : <http://bitsavers.informatik.uni-stuttgart.de/pdf/ibm/650/CarnegieInternal> Translator.pdf.

1. Kastberg P. Information and Documentation Management in the Training of Technical Translators as opposed to teaching technical science. / P. Kastberg // LSP & Professional Communication: An International Journal. – 2002. – Volume 2. – Number 1. – P. 57–66.
2. Kay M. The Proper Place of Men and Machines in Language Translation / M. Kay

// Research report CSL–80–11. – CA : Xerox Palo Alto Research Center, 1997. – P. 57–62.

1. Kirk A. Expanding Translation Use to Improve the Quality of Technical Communication / A. Kirk // Professional Communication, IEEE Transactions. – 2002. – № 43. – P. 323–326.
2. Lecarme O. More Comments on the Programming Language Pascal [Electronic resource] / O. Lecarme // Acta Informatica. – 1975. – № 4. – P. 231–243. – Access : https://docslide.com.br/documents/more-comments-on-the- programming-language-pascal.html.
3. Pasachof J. M. Scientists: Fortran vs. Modula-2 / J. M. Pasachof // BYTE (letter) ; [Retrieved 6 February 2015]. – 1984. – P. 404.
4. Sammet J. Summary of Changes in COBOL, 1960–1985 / Jean Sammet, Jerome Garfunkel // Annals of the History of Computing. IEEE. – 1985. – 7 (4). – P. 342– 345.
5. Schricker D. J4: COBOL Standardization / D. Schricker // Micro Focus. – 1998.

– P. 55–76.

1. Williams K. Grace Hopper : admiral of the cyber sea / Kathleen Williams. – Annapolis, Md : Naval Institute Press, 2004. – 340 p.

**ДОДАТОК А**

**Англо-український глосарій**

|  |  |  |
| --- | --- | --- |
| 1 аbandonware (software) | покинути, залишити програмне забезпечення | словоскладання (однокомпонентне), (еквівалентний переклад, описовий переклад) |
| 2 active matrix | активна матриця | словоскладання (двокомпонентне), абстрактна сутність,  (транскодування+калькування) |
| 3 application binary Interface | двійковий інтерфейс застосувань | словоскладання (трьохкомпонентне), абстрактна сутність,( калькування) |
| 4 Advanced Research Projects Agency | Агентство перспективних дослідницьких  проектів | словоскладання (чотирьохкомпонентне), абстрактна сутність ( калькування) |
| 5 Algoritmic Language | алгоритмiчна мова | словоскладання (двокомпонентне), абстрактна сутність,( калькування) |
| 6 arithmetic  expressions | арифметичнi  вираження | словоскладання (двокомпонентне),  абстрактна сутність ( калькування) |
| 7 arrays | массив | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 8 assignment statements | оператори присвоєння | словоскладання (двокомпонентне), абстрактна сутність ( калькування) |
| 9 augment | збiльшувати,  збільшення | однокомпонентне, (еквівалентний  переклад) |
| 10 Action Script | *скриптова мова* | словоскладання (двокомпонентне), абстрактна сутність |
| 11 аdd-in program | програмні розширення | словоскладання (трьохкомпонентне), (еквівалентний переклад) |
| 12 аperture lattice | апертурна решітка | словоскладання (двокомпонентне), абстрактна сутність (транскодування+калькування) |
| 13 boggedown in details | надмiр деталей | словоскладання (трьохкомпонентне), (еквівалентний переклад) |
| 14 body | тiло [програми] | однокомпонентне, вторинна номінація, абстрактна сутність, еквівалентний переклад |
| 15 back-end database | прикладна частина бази даних | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 16 back end software | кінцеве програмне забезпечення | словоскладання (трьохкомпонентне), абстрактна сутність (калькування) |
| 17 back-end systems | виконуючі системи | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 18 backbone | опорна мережа | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 19 backup | резервна копія | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 20 bad sector | дефектний сектор | словоскладання (двокомпонентне),  абстрактна сутність, (калькування) |
| 21 bar code | штриховий код | словоскладання (двокомпонентне), абстрактна сутність, (транскодування+калькування) |
| 22 benchmark | еталонний тест | словоскладання (однокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 23 best-of-breed solution | рішення нового покоління | словоскладання (чотирьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 24 вinding | зв’язування | однокомпонентне, суфіксація, (еквівалентний переклад) |
| 25 binary digit | двійковий знак, двійкова цифра, двійковий розряд | словоскладання (двокомпонентне), абстрактна сутність,(калькування) |
| 26 bit-map | бітовий массив | словоскладання (двокомпонентне), абстрактна сутність, (калькування + транскодування) |
| 27 browser | браузер | однокомпонентне, суфіксація, абстрактна сутність,(транскодування) |
| 28 bundled software | стандартне програмне забезпечення | словоскладання (двокомпонентне),  абстрактна сутність,( еквівалентний переклад) |
| 29 bytecode | байткод | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 30 bus mastering | одноосібне керування  шиною | словоскладання (двокомпонентне),  суфіксація, (еквівалентний переклад) |
| 31 constant | константа | однокомпонентне, абстрактна сутність, (транскодування) |
| 32 control flow | керування струмом | словоскладання (двокомпонентне), вторинна номінація (калькування) |
| 33 compile | компiлювати | однокомпонентне, (транскодування) |
| 34 computation | обчислення | однокомпонентне, суфіксація, абстрактна сутність, (автохтонний переклад) |
| 35 compiler | компiлятор | однокомпонентне, суфіксація, абстрактна сутність,(транскодування) |
| 36 character string | строка символiв | словоскладання (двокомпонентне), абстрактна сутність ( калькування) |
| 37 command interpreter | командний інтерпретатор, командний процесор | словоскладання (двокомпонентне), абстрактна сутність (транскодування+калькування) |
| 38 country code | код країн | словоскладання (двокомпонентне),  абстрактна сутність (транскодування+калькування) |
| 39 cycle | цикл | однокомпонентне, абстрактна сутність, (транскодування) |

|  |  |  |
| --- | --- | --- |
| 40 сable мodem | кабельний модем | словоскладання (двокомпонентне), предмет,(транскодування+калькування) |
| 41 card reader | кардрідер | словоскладання (двокомпонентне), суфіксація, предмет, (калькування +  транскодування) |
| 42 careware | програмне забезпечення | словоскладання (однокомпонентне), абстрактна сутність (еквівалентний переклад) |
| 43 сell | ділянка, клітина | однокомпонентне, абстрактна сутність,  (еквівалентний переклад) |
| 44 сhip | чіп | однокомпонентне, предмет, (транскодування) |
| 45 chipset | набір мікросхем | словоскладання (двокомпонентне), предмет, (еквівалентний переклад) |
| 46 chunker | програма- фрагментатор | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 47 сircuit switching | комутація каналів | словоскладання (двокомпонентне), суфіксація, абстрактна сутність,  (еквівалентний переклад) |
| 48 command line interface | інтерфейс командного рядка | словоскладання (трьохкомпонентне), абстрактна сутність, (транскодування+калькування) |
| 49 coaxial cable | коаксіальний кабель | словоскладання (двокомпонентне), предмет, (калькування +  транскодування) |
| 50 code page | кодова сторінка | словоскладання (двокомпонентне), абстрактна сутність (калькування + транскодування) |
| 51 common ground | загальна основа | словоскладання (двокомпонентне),  (калькування) |
| 52 communicator | комунікатор | однокомпонентне, суфіксація, предмет, (транскодування) |
| 53 componentware | компонентне програмне  забезпечення | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 54 concept search | концептуальний пошук | словоскладання (двокомпонентне), абстрактна сутність, (калькування) |
| 55 configuration file | файл конфігурації | словоскладання (двокомпонентне), абстрактна сутність,  (транскодування+калькування) |
| 56 dashboard | інструментальна панель | словоскладання (однокомпонентне), абстрактна сутність,( калькування) |
| 57database | база даних | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 58 data-based knowledge | знання, побудовані на даних | словоскладання (трьохкомпонентне), (еквівалентний переклад) |
| 59 database engine | процесор бази даних | словоскладання (двокомпонентне), абстрактна сутність,( еквівалентний  переклад) |

|  |  |  |
| --- | --- | --- |
| 60 datagram | дейтаграмма | словоскладання (однокомпонентне), абстрактна сутність,(транскодування) |
| 61 datasets | набори даних | словоскладання (однокомпонентне), абстрактна сутність (еквівалентний  переклад) |
| 62 decibel | децибел | однокомпонентне, абстрактна сутність, (транскодування) |
| 63 debugger | налагоджувач | однокомпонентне, суфіксація+префіксація, абстрактна  сутність, (еквівалентний переклад) |
| 64 Debugging | налагодження | однокомпонентне,  суфіксація+префіксація, (еквівалентний переклад) |
| 65 drawback | недолiк | словоскладання (однокомпонентне),  (еквівалентний переклад) |
| 66 desktop | настільний | словоскладання (однокомпонентне), (еквівалентний переклад) |
| 67dial-up connection | комутоване з’єднання по телефонній лінії | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 68 digital | цифровий, електронний | однокомпонентне, (автохтонний відповідник) |
| 69 distributed computing | розподілені обчислення | словоскладання (двокомпонентне), абстрактна сутність, (калькування) |
| 70 document file icon | піктограма файла документа | словоскладання (трьохкомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 71 domain | домен | однокомпонентне, абстрактна сутність, (транскодування) |
| 72 download | завантаження | словоскладання (однокомпонентне),  (автохтонний переклад) |
| 74 extrapolate | екстраполювати | однокомпонентне, префіксація, (транскодування) |
| 75 escape sequence | послiдовнiсть символiв | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 76 element | елемент | однокомпонентне, абстрактна сутність, (транскодування) |
| 77 encryption | шифрування | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний  переклад) |
| 78 end system | кінцева система | словоскладання (двокомпонентне), абстрактна сутність, (калькування) |
| 79 entity | сутність | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 80 enviroument | середовище | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 81 framework | рамки | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 82 feature | просторовий елемент | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 83 fiber | оптичне волокно | однокомпонентне, предмет, (автохтонний відповідник) |
| 84flash memory | флеш-пам’ять | словоскладання (двокомпонентне), абстрактна сутність, (калькування+транскодування) |
| 85 frame | кадр | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 86 free software | вільне програмне забезпечення | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 87freeware | безкоштовне програмне  забезпечення | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 88front end | зовнішня частина | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 89fuzzy theory | нечітка логіка | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 90 Gbyte | гігабайт | однокомпонентне, абревіація, префіксація, абстрактна сутність, (транскодування) |
| 91 generic protocols | протоколи, що настроюються | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 92GigaHertz | гігагерц | словоскладання (двокомпонентне), префіксація, абстрактна сутність,  (транскодування) |
| 93Grid-computing | Grid-обчислення | словоскладання (двокомпонентне), суфіксація, абстрактна сутність (транс кодування+калькування) |
| 94groupware | групове програмне забезпечення | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад+транскодування) |
| 95hack | злом | однокомпонентне, (автохтонний відповідник) |
| 96halftone | півтон | словоскладання (однокомпонентне), абстрактна сутність, (калькування) |
| 97handshaking | квитирування | словоскладання (однокомпонентне), суфіксація, (еквівалентний переклад) |
| 98hаrd disk | жорсткий диск | словоскладання (двокомпонентне), предмет, (калькування) |
| 99hard drive | накопичувач на  жорсткому диску | словоскладання (двокомпонентне),  предмет, (еквівалентний переклад) |
| 100hardware platform | апаратна платформа | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 101headend | головний вузол комп’ютерної мережі | словоскладання (однокомпонентне),  абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 102host | головний вузол | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 103hot keys | гарячі клавіші | словоскладання (двокомпонентне), предмет, вторинна номінація,  (калькування) |
| 104hot spot | точка безпровідного публічного доступу | словоскладання (двокомпонентне), абстрактна сутність, вторинна номінація, (еквівалентний переклад) |
| 105 hot swap | гаряча заміна | словоскладання (двокомпонентне),  вторинна номінація, (еквівалентний переклад) |
| 106 hypercube | гіперкуб | Словоскладання (однокомпонентне),префіксація, абстрактна сутність, (транскодування) |
| 107hypermedia | гіпермедіа | словоскладання (однокомпонентне),  префіксація, абстрактна сутність, (транскодування) |
| 108 hypertext | гіпертекст | словоскладання (однокомпонентне), префіксація, абстрактна сутність,  (транскодування) |
| 109 hertz | герц | однокомпонентне, абстрактна сутність, (транскодування) |
| 110 Input | ввід, вхідні дані | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 111 internal mode | внутрішній режим | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 112 integer | цiле число | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 113 information systems | інформаційна система | словоскладання (двокомпонентне), абстрактна сутність, (калька + транскодування) |
| 114 information technology | інформаційна технологія | словоскладання (двокомпонентне), абстрактна сутність, (калька +  транскодування) |
| 115 Infranet | Інфранет | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 116 infrared | інфрачервоний | словоскладання (однокомпонентне), абстрактна сутність, (транскодування +  еквівалент) |
| 117 instance | екземпляр | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 118 integrated circuit | інтегральна схема | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 119 Intel | інтегральна електроніка | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 120 internal network | внутрішня мережа | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад),(калькування) |

|  |  |  |
| --- | --- | --- |
| 121 internal web | внутрішня павутина | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад+кальування) |
| 122 internet | iнтернет | словоскладання (однокомпонентне),  абстрактна сутність, (транскодування) |
| 123 Internet address | iнтернет адреса | словоскладання (двокомпонентне), абстрактна сутність, (калькування+ транскодування) |
| 124 Internet protocol | протокол Інтернету | словоскладання (двокомпонентне),  абстрактна сутність, (транскодування+калькування) |
| 125 internetting | міжмережна взаємодія | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 126 internetwork | об’єднана мережа | словоскладання (однокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 127 Іntranet | Інтранет | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 128 Jini Technology | Jini-технологія | словоскладання (двокомпонентне), абстрактна сутність, (калькування + транскодування) |
| 129 just-in-time compiler | компілятор  «моментальної», трансляції | словоскладання (чотирьохкомпонентне), суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 130 just-in-time printing | роздруківка «точно в строк» | словоскладання (чотирьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 131 kilobit | kілобіт | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 132 kilobyte | кілобайт | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 134 kilobits per second | кілобіт за секунду | словоскладання (трьохкомпонентне), абстрактна сутність, (калька + транскодування) |
| 135 Knowledge Based- System | система, що ґрунтується на знаннях | словоскладання (трьохкомпонентне), абстрактна сутність, (описовий переклад) |
| 136 Knowledge Discovery in Databases | виявлення знань у базах даних | словоскладання (трьохкомпонентне), (еквівалентний переклад) |
| 137 Knowledge Engineering | інженерія знань | словоскладання (двокомпонентне),  суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 138 kilohertz | кілогерц | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 139 kit | набір, комплект | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 140 knowbot | робот знань | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |

|  |  |  |
| --- | --- | --- |
| 141 knowledge | база знань | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 142 knowledge  engineer | інженер зі знань,  когнітолог | словоскладання (двокомпонентне),  особа, (еквівалентний переклад) |
| 143 knowledge engineering | інженерія знань | словоскладання (двокомпонентне), абстрактна сутність суфіксація, (еквівалентний переклад) |
| 144 knowledge management | керування знаннями | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 145 Local Area Network | локальна обчислювальна мережа | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 146 laptop | лептоп | словоскладання (однокомпонентне), предмет, (транскодування) |
| 147 layer | рівень, шар | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 148 layout | компонування карти | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 149 layout grid | макет сторінки | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 150 Liquid Crystal Display | дисплей, рідкокристалічний дисплей | словоскладання (трьохкомпонентне), предмет, (еквівалентний переклад), (транс кодування) |
| 151 Luminance | яскравість | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 152 lead-in | початкова зона на диску | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 153 lead-out | кінцева зона на диску | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 154 leased line | виділена лінія | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 155 Light-Emitting Diode | світлодіодний індикатор | словоскладання (трьохкомпонентне), предмет, (еквівалентний переклад) |
| 156 legacy system | успадкована система | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад), (калькування) |
| 157 Last In First Out | останнім увійшов, першим вийшов | словоскладання (чотирьохкомпонентне), (еквівалентний переклад) |
| 158 lightscribe | оптичне гравірування | словоскладання (однокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 159 Link | з’єднання, канал зв’язку, зв’язок | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 160 Locator Type Element | елемент типу локатор | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 161 Login account | обліковий запис реєстраційного імені | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 162 login name | входити в систему | словоскладання (двокомпонентне), (еквівалентний переклад) |
| 163 line per inch | ліній на дюйм | словоскладання (трьохкомпонентне),  абстрактна сутність, (еквівалентний переклад), (калькування) |
| 164 line per second | ліній за секунду | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад), (калькування) |
| 165 Line Printer | порядковий принтер | словоскладання (двокомпонентне), предмет, (калькування+транскодування) |
| 166 Literal Result Element | літеральний елемент результату | словоскладання (трьохкомпонентне), абстрактна сутність, (калькування+транскодування) |
| 167 Media Access Control | керування доступом до середовища передавання | словоскладання (трьохкомпонентне), (еквівалентний переклад) |
| 168 macro | макрокоманда, макрос | однокомпонентне, абстрактна сутність, (транскодування), (транскодування) |
| 169 mainframe | мейнфрейм | словоскладання (однокомпонентне),  префіксація, абстрактна сутність, (транскодування) |
| 170 maintainability | ремонтопридатність | словоскладання (однокомпонентне), префіксація, абстрактна сутність,  (еквівалентний переклад) |
| 171 maintainer | мейнтейнер | словоскладання (однокомпонентне), префіксація, абстрактна сутність, (транскодування) |
| 172 maintenence | супровід | словоскладання (однокомпонентне),  +префіксація, абстрактна сутність,  (еквівалентний переклад) |
| 173 malware | шкідливе ПЗ | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 174 Metropolitan Area Network | міська  (обчислювальна) мережа | словоскладання (трьохкомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 175 managed code | код керований | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 176 manifest | маніфест | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 177 map | карта | однокомпонентне, вторинна номінація,  абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 178 mapping | відображення | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 179 marshaling | маршалінг, транспортування | однокомпонентне, суфіксація,  (еквівалентний переклад), (транскодування) |
| 180 Multiаgent Systems | мультиагентна система | словоскладання (двокомпонентне), префіксація, абстрактна сутність, (еквівалентний переклад) |
| 181 Mathematical Markup Language | математична мова розмітки | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 182 megabyte | мегабайт | словоскладання (однокомпонентне), префіксація, абстрактна сутність,  (транскодування) |
| 183 megabit | mегабіт | словоскладання (однокомпонентне), префіксація, абстрактна сутність,  (транскодування) |
| 184 MegaBytes per second | мегабайт за секунду | словоскладання (трьохкомпонентне),  префіксація, абстрактна сутність, (калькування+транскодування) |
| 185 Megabits per second | мегабіт за секунду | словоскладання (трьохкомпонентне), префіксація, абстрактна сутність, (калькування+транскодування) |
| 186 Media Control Interface | інтерфейс керування носіями | словоскладання (трьохкомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 187 Mail Delivery Agent | агент доставляння електронної пошти | словоскладання (трьохкомпонентне), абстрактна сутність,  (калькування+транскодування) |
| 188 Mobile Document Access | доступ до мобільних документів | словоскладання (трьохкомпонентне), абстрактна сутність, (калькування+транскодування) |
| 189 Model Driven Architecture | архітектура, керована моделлю | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 190 Multiple-Document Interface | багатодокументний інтерфейс | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 191 Memory Stick | формати карт пам’яті | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 192  Microelectromechanical systems | мікроелектромеханічні системи | словоскладання (двокомпонентне),префіксація, абстрактна сутність,  (калькування+транскодування) |
| 193 Manufacturing Execution Systems | системи керування виробництвом | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 194 Manufacturing Enterprise Solutions | рішення для  виробничих підприємств | словоскладання (трьохкомпонентне), (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 195 micro | мікро | однокомпонентне, абстрактна сутність, (транскодування) |
| 196 microcomputer | мікрокомп’ютер | словоскладання (однокомпонентне), префіксація, предмет,  (калькування+транскодування) |
| 197 microprocessor | мікропроцесор | словоскладання (однокомпонентне), префіксація,предмет, (транскодування) |
| 198 microsecond | мікросекунда | словоскладання (однокомпонентне), префіксація, абстрактна сутність  (транскодування) |
| 199 Mobile Internet Device | мобільний інтернет- пристрій | словоскладання (трьохкомпонентне), предмет(калькування+транскодування) |
| 200 Module Interconnection  Language | мова взаємодії модулів | словоскладання (трьохкомпонентне), абстрактна сутність,  (калькування+транскодування) |
| 201 mirror | дзеркальне відображення | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 202 mirroring | дзеркальне копіювання | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний  переклад) |
| 203 mobile computer | кишеньковий комп’ютер | словоскладання (двокомпонентне), предмет, (еквівалентний переклад) |
| 204 mode | модальний | Однокомпонентне, (транс кодування) |
| 205 mobile device | мобільний пристрій | словоскладання (двокомпонентне), предмет, (еквівалентний переклад) |
| 206 modeless | немодальний | однокомпонентне, (еквівалентний  переклад) |
| 207 modulation | модуляція | однокомпонентне, суфіксація, абстрактна сутність, (транскодування) |
| 208 morphing | морфінг | однокомпонентне, абстрактна сутність, (транскодування) |
| 209 motherboard | материнська плата | словоскладання (однокомпонентне), предмет, (еквівалентний переклад) |
| 210 most significant byte | найбільш значимий біт або байт | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 211 ms | мілісекунда | словоскладання (однокомпонентне) абревіація, префіксація, абстрактна сутність, (транскодування) |
| 212 Mean Time Between Failure | час між відмовами | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 213 multi-homed computer | багатомережний комп’ютер | словоскладання (трьохкомпонентне), префіксація, предмет, (еквівалентний переклад) |
| 214 multi-homed host | багатомережний хост | словоскладання (трьохкомпонентне), префіксація, абстрактна сутність,  (еквівалентний переклад), (транскодування) |

|  |  |  |
| --- | --- | --- |
| 215 multicasting | мультимовлення | словоскладання (однокомпонентне), префіксація, абстрактна сутність, (транскодування) |
| 216 multinetting | множинне зв’язування | словоскладання (однокомпонентне),  суфіксація+префіксація, абстрактна сутність, (еквівалентний переклад) |
| 217 мultiplexer | мультиплексор | однокомпонентне, суфіксація+префіксація, предмет,(транскодування) |
| 218 multi-touch screen | сенсорний екран | словоскладання (трьохкомпонентне),префіксація,  предмет, (еквівалентний переклад) |
| 219 N-tier application | N-ярусне застосування | словоскладання (трьохкомпонентне), суфіксація, абстрактна сутність,  (калькування) |
| 220 name resolution | розрізнення імен | словоскладання (двокомпонентне), суфіксація, абстрактна сутність,  (еквівалентний переклад) |
| 221 naming | іменування | однокомпонентне, суфіксація,  абстрактна сутність, (еквівалентний переклад) |
| 222 nanosecond | наносекунда | словоскладання (однокомпонентне), абстрактна сутність, (транскодування) |
| 223 narrowband | вузька смуга пропускання | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 224 Network Address Translation | трансляція мережних адрес | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 225 Network Basic Input/Output System | мережна базова система вводу-виводу | словоскладання (трьохкомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 226 netbook | нетбук | словоскладання (однокомпонентне), предмет, (транскодування) |
| 227 Netware | Мережна операційна система | словоскладання (однокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 228 network | мережа | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 229 network bridge | мережний міст | словоскладання (двокомпонентне),  абстрактна сутність, (еквівалентний переклад), (калькування) |
| 230 network center | мережний центр | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 231 network operating system | *мережна операційна система* | словоскладання (трьохкомпонентне), суфіксація, абстрактна сутність,  (еквівалентний переклад), (калькування) |
| 232 Network File System | мережна файлова система | словоскладання (трьохкомпонентне), абстрактна сутність, (калькування) |

|  |  |  |
| --- | --- | --- |
| 233 Next Generation Network | мережа зв’язку наступного покоління | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 234 Network Interface  Card | мережний адаптер | словоскладання (трьохкомпонентне),  предмет, (еквівалентний переклад) |
| 235 node | вузол мережі | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 236 northbridge | північний міст | словоскладання (однокомпонентне), предмет, (еквівалентний переклад),  (калькування) |
| 237 notebook | ноутбук | словоскладання (однокомпонентне), предмет, (транскодування) |
| 238 notepad computer | комп’ютер у вигляді записника | словоскладання (двокомпонентне), предмет, (еквівалентний переклад) |
| 239 number cruncher | перемелювання чисел | словоскладання (двокомпонентне), вторинна номінація, (еквівалентний переклад) |
| 240 Object Browser | засіб перегляду об’єктів | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 241 obstruction | перешкода, бар’єр | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 242 Optical Character Recognition | оптичне розпізнавання символів | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 243 off-line | oфф-лайн – автономний | словоскладання (двокомпонентне), (транскодування) |
| 244 on-line | *oн-лайн* | словоскладання (двокомпонентне), (транскодування) |
| 245 on the fly | на льоту | словоскладання (двокомпонентне) (транскодування) |
| 246 optical cable | волоконно-оптичний кабель | словоскладання (двокомпонентне), предмет, (автохтонний відповідник) |
| 247 optical fibre | оптоволокно | словоскладання (двокомпонентне),  предмет, (автохтонний відповідник) |
| 248 orchestration | оркестровка | однокомпонентне, суфіксація, (транскодування) |
| 240 orgware | організаційний продукт | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 250 Open Systems Interconnection | взаємодія відкритих систем | словоскладання (трьохкомпонентне), суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 251 overclocker | *оверклокер* | однокомпонентне, суфіксація,  абстрактна сутність, (транскодування) |
| 252 ontology language | онтологічна мова для веб | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 253 package | пакет | однокомпонентне, вторинна номінація, абстрактна сутність, (еквівалентний переклад) |
| 254 packet switching | пакетна комутація | словоскладання (двокомпонентне), суфіксація, вторинна номінація,  абстрактна сутність, (еквівалентний переклад) |
| 255 page layout | процес верстки сторінки | словоскладання (двокомпонентне), вторинна номінація, абстрактна сутність,  (еквівалентний переклад) |
| 256 palmtop | *надолонний комп’ютер* | словоскладання (однокомпонентне), предмет, (еквівалентний переклад) |
| 257 paradigm | *парадигма* | однокомпонентне, абстрактна сутність, (транскодування) |
| 258 parent element | елемент батьківський | словоскладання (двокомпонентне),  вторинна номінація, абстрактна сутність, (еквівалентний переклад) |
| 259 parity | парність | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 260 parser | синтаксичний аналізатор | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 261 parsed character data | символьні дані що піддаються  синтаксичному аналізу | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 262 parsed entity | сутність, що піддається синтаксичному аналізу | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 263 password | *пароль* | словоскладання (однокомпонентне),  абстрактна сутність, (еквівалентний переклад) |
| 264 patch cable | з’єднувальний шнур | словоскладання (двокомпонентне), предмет, (еквівалентний переклад) |
| 265 pattern | *патерн, шаблони*  *проектування* | словоскладання (однокомпонентне),  абстрактна сутність, (транскодування) |
| 266 PByte | *петабайт* | словоскладання (однокомпонентне),абревіація, абстрактна сутність, (транскодування) |
| 267 Personal Computer | персональний комп’ютер | словоскладання (двокомпонентне), предмет, (транскодування+калькування) |
| 268 physical document structure | структура документа фізична | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 269 piggybacking | комплексна дія | словоскладання (однокомпонентне),  суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 270 playlist | список файлів для програвання | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 271 plug | рознім | однокомпонентне, предмет, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 272 plug-in | плагін | словоскладання (двокомпонентне), абстрактна сутність, (транскодування) |
| 273 portable | портабельний | однокомпонентне, (транскодування) |
| 274 portable code | портабельний код | словоскладання (двокомпонентне), абстрактна сутність,  (транскодування+калькування) |
| 275 post-gap | після-пауза | словоскладання (двокомпонентне), абстрактна сутність, (еквівалениний переклад+калькування) |
| 276 postmaster | фахівець з питань  керування поштовими сервісами | словоскладання (однокомпонентне), особа, (описовий переклад) |
| 277 print screen | друк екрана | словоскладання (двокомпонентне), абстрактна сутність, (калькування) |
| 278 proxy | проксі, функція- заступник | однокомпонентне, абстрактна сутність,  (транскодування, автохтонний відповідник) |
| 279 proxy-server | проксі-сервер | словоскладання (двокомпонентне), абстрактна сутність, (транскодування+калькування) |
| 280 public-domain software | програми, що вільно копіюються | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 281quantum computing | *квантові обчислення* | словоскладання (двокомпонентне), суфіксація, абстрактна сутність(транскодування+калькування) |
| 282quantum bit | квантовий біт | словоскладання (двокомпонентне), абстрактна сутність, (транскодування+калькування) |
| 283query | запит | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 284radio access network | мережа радіодоступу | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 285 read-only | тільки для читання | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 286 real-time language | *мова реального часу* | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 287receipting | *квитирування* | однокомпонентне, суфіксація,  абстрактна сутність, (еквівалентний переклад) |
| 288 removability | переміщуваність | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 289 rendered document | наведений документ | словоскладання (двокомпонентне), абстрактна сутність,(транскодування+калькування) |
| 290 resolution | *розрізнення* | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |

|  |  |  |
| --- | --- | --- |
| 291 resource type element | елемент типу ресурс | словоскладання (трьохкомпонентне), абстрактна сутність, (еквівалентний переклад) |
| 292rootkit | *руткит* | словоскладання (однокомпонентне),  абстрактна сутність,(транскодування) |
| 293router | *маршрутизатор* | однокомпонентне, абстрактна сутність, (еквівалентний переклад) |
| 294 run time | час прогону | словоскладання (двокомпонентне), вторинна номінація, абстрактна сутність,  (калькування) |
| 295 sandbox | пісочниця | словоскладання (однокомпонентне),  вторинна номінація, абстрактна сутність, (калькування) |
| 296 sandboxing | завантаження у “пісочницю | словоскладання (однокомпонентне), суфіксація, абстрактна сутність, (калькування) |
| 297 schema | *схема* | однокомпонентне, абстрактна сутність,  (транскодування) |
| 298 screen saver | зберігач екрана, скринсейвер | словоскладання (двокомпонентне), абстрактна сутність,  (транскодування+калькування) |
| 299 scripting | створення сценаріїв, скриптова мова | однокомпонентне, суфіксація,  абстрактна сутність (еквівалентний переклад) |
| 300 sharpness | різкість | однокомпонентне, суфіксація, абстрактна сутність, (еквівалентний переклад) |
| 301 shell | *командний процессор* | однокомпонентне, вторинна номінація, абстрактна сутність, (еквівалентний  переклад) |
| 302 shortcut menu | оперативне меню | словоскладання (двокомпонентне), абстрактна сутність,  (транскодування+калькування) |
| 303 side-by-side execution | паралельне виконання программ | словоскладання (чотирьохкомпонентне), абстрактна сутність |
| 304 slot | *слот* | однокомпонентне, абстрактна сутність,  (транскодування) |
| 305 softlifting | розмноження ПЗ, софтліфтинг | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 306 software piracy | програмне піратство | словоскладання (двокомпонентне),  абстрактна сутність, (калькування+еквівалентний переклад) |
| 307 software process | програмний процесс | словоскладання (двокомпонентне), абстрактна сутність,(транскодування+калькування) |
| 308 solution | *рішення* | однокомпонентне, абстрактна сутність,  (еквівалентний переклад) |
| 309 spam | спам | однокомпонентне, абстрактна сутність, (транскодування) |

|  |  |  |
| --- | --- | --- |
| 310 spyware | шпигунське ПЗ | словоскладання (однокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 311 stakeholder | організатор | словоскладання (однокомпонентне), особа, (автохтонний переклад) |
| 312 stacker | стекер | однокомпонентне, суфіксація, абстрактна сутність, (транскодування) |
| 313 standby mode | резервний режим | словоскладання (двокомпонентне), абстрактна сутність, (еквівалентний  переклад) |
| 314 star schema | схема «зірка» | словоскладання (двокомпонентне),  вторинна номінація, абстрактна сутність, (транскодування+калькування) |
| 315 startup disk | системна дискета | словоскладання (двокомпонентне), предмет, (транскодування+калькування) |
| 316 steganography | *стеганографія* | словоскладання (однокомпонентне), абстрактна сутність,(транскодування) |