

Ю.О. Ющенко

РОЗРОБКА АРХІТЕКТУРИ КОМП'ЮТЕРА «КИЇВ» ЗА КОНЦЕПЦІЄЮ АДРЕСНОГО МЕТОДУ ПРОГРАМУВАННЯ

Описано поступовий перехід від розрахунків логарифмічними лінійками та арифмометрами до використання мови високого рівня з вказівниками та складними ієрархічними структурами. В статті розглянуто фактори, які сприяли цьому важливому технологічному переходу. Робота відновлює загублену ланку історії виникнення в Україні опосередкованої адресації (вказівників) вищих рангів та складних ієрархічних структур. Надаються підтвердження винайдення вказівників та складних структур даних київськими вченими, описано апаратну реалізацію в комп'ютері «Київ» «штрих-операції» (розмінування вказівників) та операцій задання циклів. У роботі обґрунтовується високорівневість програмування у командах комп'ютера «Київ» шляхом порівняння з мовою програмування високого рівня – Plankalkül.

Ключові слова: історія, штрих-операція, опосередкована адресація, вказівники, програмування, деревоподібні формати, масиви, списки, структури, абстрактні типи даних.

Вступ

Наразі спостерігається підвищення цікавості людей до історії інформаційних технологій (ІТ), що обумовлено лавиноподібним зростанням значущості ІТ у всіх сферах науки, виробництва, економіки, політики та взагалі у всіх видах діяльності та дозвілля людей. У світі особлива увага приділяється зародженню інформаційних технологій.

Зародженню перших комп'ютерів та виникнення програмування присвячено багато публікацій. Однак, на думку автора, втрачена важлива ланка в історії ІТ, що б мала пояснити перехід від низькорівневого програмування до високорівневого. Стаття, на основі наукових публікацій [2-5, 7, 11, 10, 14, 20, 21, 26, 29, 39-42], спогадів очевидців [12, 13, 15, 18, 19, 22, 23, 30, 31, 38, 46], історичних публікацій [9, 24, 32-34] та спогадів К.Л. Ющенко, відновлює цю загублену ланку.

Мета статті відновити загублену в історії ланку зародження в Україні програмування високого рівня з потужними засобами, які використовуються в усіх сучасних технологіях програмування. Метою статті є визначення факторів, які сприяли винайденню високорівневого Адресного програмування з потужними можливостями адресації.

На зародження в Україні високорівневого програмування вплинуло використання інструкцій для розрахунків арифмометрами та досвід програмування на комп'ютері МЕСМ¹ [11 (стор. 36: Т. 2), 18, 43] з обмеженими ресурсами. Під час розробки асинхронного комп'ютера «Київ» важливу роль відіграли висококваліфіковані інженери, які розроблювали МЕСМ [15].

Виникненню програмування високого рівня сприяв операторний метод програмування [11], до якого О.А. Ляпунов прийшов під час досліджень програмування на МЕСМ у Феофанії влітку 1952р. [44]. 1953 року в наукових статтях, семінарах та лекціях О.А. Ляпунов оприлюднює операторний метод програмування [9, 11, 18, 21, 22, 29, 43].

Можливість динамічної модернізації програм МЕСМ наштовхнула на ідею створення однією програмою інших програм, тобто програмуючих програм, які згодом було названо компіляторами та трансляторами.

1. Діяльність Б.В. Гнеденка

Б.В. Гнеденко 1949 року, на семінарі із розробки МЕСМ, порадив С.О. Лебедеву збільшити розрядність комірок пам'яті з 12 до 17 [13, 18, 23, 24, 30]. Без цього збіль-

¹ Російською “МЭСМ” (малая электронная счетная машина).

шення комп'ютер не був би придатний для розрахунків.

Б.В. Гнеденко запропонував долучитися до програмування В.С. Королюку, який під час навчання в аспірантурі у А.М. Колмогорова, відвідував московські наукові семінари із програмування та слухав лекції О.А. Ляпунова [13, 18].

Після повернення з НДР Б.В. Гнеденко очолює лабораторію обчислювальної техніки, яка утворюється шляхом об'єднання обчислювальної лабораторії ІМ УН УРСР (керівник К.Л. Ющенко) [31] з лабораторією моделювання та обчислювальної техніки Інституту електродинаміки (керівник С.О. Лебедев).

В.С. Королюк разом із К.Л. Ющенко з 1955/56р. навчального року починають читати лекції у КДУ та КПІ по операторному та адресному методах програмування, організованих Б.В. Гнеденком [13, 23, 46]. 1957 року К.Л. Ющенко та В.С. Королюк опублікували у КДУ обмеженим накладом перший у СРСР підручник із програмування з описом операторного та адресного методів програмування [46 (стор. 36)]. Цей підручник побачив світ 1961р. та перекладався й видавався у європейських країнах: Угорщині [5], Чехословаччині, східній Німеччині [4], Югославії та Данії. 1963р. додатки до цього підручника з описом архітектури п'яти радянських комп'ютерів, включаючи комп'ютер «Київ», перекладаються англійською мовою та публікуються у США 1963р. [7]. А 1969р., після 12 років від часу написання, видають французькою у Франції [3]. У підручнику описується концепція опосередкованої адресації (Pointers) вищих рангів, відношення слідування, а у додатку з описом архітектури комп'ютера «Київ» детально описано операції модифікації адрес із можливістю виконання «штрих-операції» (розіменування Pointers) та операцій задання циклів.

Очолюючи лабораторію обчислювальної техніки, Б.В. Гнеденко організовує наукові семінари із проектування архітектури великого універсального асинхронного керуючого комп'ютера «Київ» та автоматизації програмування. Як науковий

керівник розробки комп'ютера заглиблювався в деталі архітектури та тонкощі програмування [13, 23, 24, 46].

2. Актуальність

У світі приділяється велика увага історії зародження програмування, створено багато музеїв історії інформаційних технологій, існує багато публікацій, знято низку документальних та художніх фільмів.

На підставі детального дослідження та аналізу доступних джерел автор дійшов висновку, що важлива ланка переходу від програмування низького рівня до програмування високого рівня [9, 11, 12, 18, 21, 22, 30, 38] не вповні описана в історії ІТ. Лише частково цей перехід заповнює операторний метод програмування, який по праву вважається передумовою зародження програмування високого рівня у СРСР.

В Україні перехід від програмування низького рівня до високорівневого програмування відбувався одночасно з включенням в нього опосередкованої адресації (Pointers) вищих рангів та універсальних засобів групування даних і підпрограм у складні ієрархічні структури (деревоподібні формати²). Основний науковий результат цієї статті полягає у визначенні передумов та причинно-наслідкових зв'язків, які сприяли та привели до винаходу киянами потужних засобів програмування, без яких не обходиться жодна сучасна технологія програмування. Однак історики інформаційних технологій ніколи не зазначають про цей значний внесок українців у зародження та розвиток програмування.

Історія та передумови виникнення найпотужніших засобів у програмуванні дуже цікава програмістам, але досі залишалася не розкритою.

Важливість статті полягає у детальному аналізі передумов поступового зародження в Україні програмування високого рівня з вказівниками та універсальними можливостями групування та з'єднання даних і підпрограм у деревоподібні формати (складні ієрархічні структури), яким подібні абстрактні типи даних.

² В оригінальних працях [40] використовується термін «деревеообразные форматы» (рос.).

Чомусь, ані державні структури, ані відомі ІТ-компанії не виявляють зацікавлення у визнанні світовою спільнотою важливих досягнень українців в ІТ. Українські ЗМІ, музеї та інші державні та недержавні установи мають приділяти гідну увагу внеску українських вчених у розвиток світових ІТ. Україна має донести, як до своїх співвітчизників, так і до міжнародної спільноти, інформацію про всесвітньо значущі досягнення українців.

Світова спільнота докладає багато зусиль для збереження історії зародження та розвитку ІТ. Визнання всім світом суттєвого внеску українців у зародження потужних засобів програмування має велике значення для збереження всесвітньої історії ІТ та дуже важливе для національно патріотичного виховання молоді.

3. Публікації по Адресному програмуванню

Існує багато підтверджень того, що саме в Феофанії у 50-х роках минулого сторіччя були винайдені фундаментальні засоби універсального групування, з'єднання даних та підпрограм у складні ієрархічні структури. Зазначені підтвердження наявні у публікаціях із Адресної мови програмування, із архітектури комп'ютера «Київ» та в інших джерелах, зокрема, в [11, 20]. Підручник із програмування, який публікувався багатьма мовами, містить опис опосередкованої адресації (вказівників) вищих рангів. У додатках до підручника містяться підтвердження апаратної реалізації в комп'ютері «Київ» операцій маніпуляцій з адресами 2-ого рангу (вказівниками) та задавання циклів. Окрім підручника з додатками вичерпні та беззаперечні підтвердження винаходу українцями вказівників опубліковані 1966р. у США [2] в перекладеній англійською мовою монографії, присвяченій комп'ютеру «Київ» з описом Адресної мови програмування [10].

Однак, у закордонних джерелах з історії програмування не згадуються випередження українцями світових досягнень на багато років. Українським програмістам також мало відомо про Адресну мову, а про те, що в ній були вказівники, складні ієрар-

хічні структури та декларативні можливості [1-5, 11, 10, 20, 39-42, 45] вони не вірять.

Головні підтвердження винаходу українськими вченими вказівників та складних ієрархічних структур наявні в архітектурі асинхронного комп'ютера «Київ». Так, його Ф-операція може виконувати «штрих-операції» (розіменування Pointers), а групові операції дозволяють задавати цикли [9 (Т. 1 стор. 489)]. При цьому мови програмування ФОРТРАН, КОБОЛ та АЛГОЛ-60, які помилково вважаються першими мовами програмування високого рівня, з'являються декількома роками пізніше за Адресну мову.

Опублікований у США 1963р. опис архітектури комп'ютера «Київ» [7] свідчить, що американським ученим у 1964р., на час винаходу ними вказівників (Pointers), вже було відомо про винахід киянами опосередкованої адресації (Pointers). За таких обставин можна припустити, що американські вчені досі не звернули увагу на доступні їм та наявні у США матеріали [2, 7] з Адресної мови програмування та комп'ютеру «Київ».

4. Причини невідомості

Розглянемо причини, які пояснюють, чому випередження українців невідомі за кордоном та маловідомі в самій Україні.

Перша причина полягає у тому, що до 1956р. розв'язувалися задачі для оборонної космічної галузі, включаючи розрахунки для проектування водневої бомби, наслідків її вибуху, розрахунки траєкторій польотів дальніх балістичних та космічних ракет [15, 18, 30, 31]. Ці роботи були під грифом суворої державної таємниці, що було перепорою в оприлюдненні будь-яких матеріалів.

Друга причина криється у завантаженості МЕСМ. Загальновідомо, що у 1952/53 роках МЕСМ був фактично єдиним у СРСР комп'ютером, який регулярно використовувався для розв'язку задач [15 (стор. 53)]. Перелік цих задач опубліковано у джерелах [18, 30, 31, 43].

Третя причина полягає у недооціненні значущості програмування у часи зародження інформаційних технологій. Законів щодо захисту прав на інтелектуальну

власність у Радянському союзі не існувало. Вироби інженерів є матеріалізованими, їх можна побачити та доторкнутися до них, а програму ні побачити, ні відчутти на дотик нема як. Оплата праці програмістів була значно меншою від зарплатні інженерів.

Четверта причина криється у тому, що за тоталітарного режиму не надто вірили, що у «провінційному» Києві, без фінансування з центру, Москви, можливі якісь суттєві досягнення. В СРСР увагу та розголос набували лише ті досягнення, які керівникам від науки вдавалося заявляти за свої [13, 24, 32].

П'ята причина полягає у тому, що тоталітарним комуністичним режимом було прийнято рішення переходу на мови ФОРТРАН, КОБОЛ та АЛГОЛ-60 та забороні подальшої роботи над Адресною мовою, яка була значно потужнішою за ці мови та мала очевидні переваги над ними. Було також заборонено публікувати та виступати на семінарах чи конференціях. Розпочаті роботи із реалізації компілятора з Адресної мови програмування для БЕСМ-6 було припинено [32, 44]. В останній публікації з Адресної мови [26] автори намагаються відстояти Адресну мову та демонструють її переваги над мовами ФОРТРАН, КОБОЛ та АЛГОЛ-60, але «вирок» було винесено. Однак, саме тоді перекладається та видається підручник з Адресної мови [14] у багатьох країнах східної Європи [4, 5] та у Франції [3], а 1966р. монографія [10] з описом архітектури комп'ютера «Київ» та Адресної мови публікується англійською у США [2].

Рано чи пізно людству стануть відомі видатні винаходи українців. Визнання значного внеску українців у технології програмування стримується певними обставинами. Багато українських програмістів не вірять або не розуміють, що вказівники є повним аналогом опосередкованої адресації 2-ого рангу. Обмаль публікацій з деревоподібних форматів та їх застосування. Закордонні програмісти повідомлення про винаходи в Україні вказівників 1955р. відносять до «фейкових» та вважають, що Україна мала достатньо часу для визнання своєї першості у таких важливих для людства винаходах.

5. Передумови зародження Адресного програмування

Вельми часто у джерелах зустрічається твердження, що перші комп'ютери розроблювались інженерами без участі математиків. Однак це не відповідає дійсності, бо 1948 року С.О. Лебедев до семінарів по проектуванню комп'ютера МЕСМ, зокрема, для визначення операцій, залучав видатних математиків: А.О. Дороніцина, К.А. Семендяєва, М.О. Лаврентьєва, Б.В. Гнеденка, О.Ю. Ішлінського, О.О. Харкевича та ін. [13, 18, 24]. Також у розробці комп'ютера «Київ», як і інших київських комп'ютерів, безпосередньо брали участь математики. Керівником розробки комп'ютера «Київ» був математик Б.В. Гнеденко. Інженер Л.Н. Дашевський керував розробкою інженерної складової. Розробкою архітектури та математичного (програмного) забезпечення опікувалася К.Л. Ющенко.

Розробка математичних вимог до комп'ютера «Київ» відбувалася паралельно з роботою над створенням та розвитком Адресної мови [10 (стор. 53)].

Отже, розвиток технічної бази та програмного забезпечення в Україні відбувався нерозривно один від одного. З одного боку архітектура та можливості реалізації операцій впливали на можливості та методи програмування, а з іншого – на вибір операцій вплинули особливості Адресної мови програмування. Система операцій розроблювалось з урахуванням потреб зручності програмуванні та спрямовувалась на легкість складання та сприйняття програм. До системи команд включались «скорочення» (мнемонічні коди), які застосовувались програмістами на МЕСМ [43].

Автоматизація програмування у Києві відбувалась незалежно та вельми відірвано від світового розвитку ІТ [20]. Є можливість чітко визначити особливості поступового переходу київськими математиками від розрахунків за інструкціями логарифмічними лініями та арифмометрами до автоматичних розрахунків програм мовою програмування високого рівня з опосередкованою адресацією (вказівниками) та деревоподібними форматами. Вирішальну роль у зазначеному переході віді-

грає унікальна архітектура МЕСМ, яка наявністю динамічної зміни програм підказала можливість створення програмою іншої програми (компіляторів та трансляторів) [11]. Без наявності у Феофанії МЕСМ нічого б не могло бути винайдено.

Чи могло програмування високого рівня зародитися ще до появи комп'ютера? Саме так відбувалося в Києві.

Неможливо визначити дату виникнення програмування високого рівня, оскільки зародження його відбувалось поступово та почалося ще до використання комп'ютера в інструкціях для розрахунків логарифмічними лінійками та арифмометрами.

6. Інструкції розрахунків арифмометрами

Розповсюджена думка, що на зміну програмуванню у двійкових машинних командах одразу прийшло програмування високого рівня, є хибною. Досвід складання інструкцій для проведення розрахунків з використанням арифмометрів, логарифмічних лінійок та рахівниць можна вважати першим кроком до зародження програмування високого рівня.

У 1950 році для розв'язку навігаційних задач за допомогою математичної теорії гіроскопічних систем О.Ю. Ішлінського, ним, директором математичного інституту АН УРСР, була створена лабораторія³ методів обчислень та розрахунків. Керівником лабораторії було призначено К.Л. Рвачову (після одруження - Ющенко) [31].

Математичний інститут задля полегшення проведення розрахунків придбав електронні лічильні арифмометри RheinMetal, які розмістили у підвальному приміщенні президії АН УРСР [30, 31]. Окрім електронних арифмометрів, використовувались і механічні арифмометри, логарифмічні лінійки, а для фіксації проміжних результатів і звичайні рахівниці. Також, під рукою, були таблиці значень елементарних функцій.

Математики обчислювальної лабораторії умовно розділялись на дві групи. Представники однієї групи «методів об-

числень» за завданням видатних математиків, згідно описаних ними загальних принципів розв'язку задач, формулювали завдання математикам-обчислювачам з іншої групи у вигляді **інструкцій** з детальним описом дій. Ці інструкції писались природною мовою з використанням формальних математичних позначок. Зокрема, інструкції містили формули, розгалуження з перевіркою умов, елементарні функції та математичні позначки: \forall , \exists , Σ , Π та інші. Вочевидь, що при виборі операцій МЕСМ, було враховано існуючі на той час методи проведення обчислень із цими позначками. У наступному розділі буде детально розглянуто, як такі позначки перетворювались у двійкові команди МЕСМ.

Окремі пункти інструкцій являли собою перевірку умов із визначенням переходу на попередній пункт (циклування). Обчислювальна лабораторія розросталася відповідно до потреб проведення розрахунків для різних задач народного господарства. До лабораторії відряджались математики з різних підрозділів математичного інституту.

На той час у Феофанії, передмісті Києва, тривала розробка комп'ютера МЕСМ, який планувався та призначався для використання математиками ІМ АН УРСР. Співробітники обчислювальної лабораторії отримували доступ до державної таємниці.

7. Програми на МЕСМ

Із 12.01.1952р. частину співробітників обчислювальної лабораторії, включаючи К.Л. Ющенко (Рвачову) та її чоловіка, Ющенка О.А., було відряджено до Феофанії для експлуатації МЕСМ, а частина продовжувала рахувати арифмометрами.

Усе, що стосувалося комп'ютера МЕСМ, знаходилося під грифом суворої державної таємниці.

Експлуатація МЕСМ внесла корективи у розподіл праці між існуючими групами математиків лабораторії. Відтепер з'явилась можливість перекласти частину розрахунків на комп'ютер. Для використання комп'ютера частина математиків

³ Іноді у джерелах лабораторію називають обчислювальною лабораторією. Згодом цю лабораторію перейменували у лабораторію методів обчислень та програмування.

опанувала складання програм. Обчислювачам, які «обертали» ручку арифмометра доручили вводити програми та дані до пам'яті МЕСМ.

Процес введення команд у пам'ять комп'ютера був не з простих: команди вводились по «0» та «1» шляхом вставки в отвори штекерів, опускання і піднімання тумблерів [11, 18].

Обчислювачі записували олівцем у зошити результати роботи комп'ютера з оперативної пам'яті: кожен біт оперативної пам'яті МЕСМ відображався на фанерних шафах у вигляді сяючої або згаслої лампочки. Для виконання цих робіт штат обчислювачів поповнюється студентками та дівчатами з середньою освітою [31].

Для введення програм застосовувались перфострічки. Був магнітний барабан для збереження програм та даних, а 1953р. підключають і друкарський пристрій.

Для зручності складання програм для МЕСМ математики замінюють коди операції позначками, а операнди команд мнемонічними літерами. Згодом ці мнемонічні коди були вдосконалені внесенням в них простих арифметичних виразів та шаблонів циклів [43]. Обчислювачі, виконуючи введення та виведення даних, допомагали програмістам та звільняли їх від рутинних дій [43].

Так поступово відбувалось наближення до високорівневого програмування. В документації по МЕСМ містяться приклади програм із циклами. Можливості команди складання команд МЕСМ дозволяють модифікувати команди в тілі циклів для отримання доступу до значень у сусідніх, послідовно розташованих комірках оперативної пам'яті комп'ютера, які у операторному методі програмування отримали назву «елементи масивів».

Обмеженість ресурсів МЕСМ відіграла позитивну роль, оскільки змусила програмістів знаходити вишукані прийоми програмування [11, 18].

Математичні позначки суми Σ та добутку Π у формулах за своїм математичним значенням представляють визначення дій, що повторюються ци-

клічно. Під позначками Σ та Π могли міститись елементи векторів або матриць: $A = \sum_{i=1}^n a_i$, або не міститись:

$$e^x = \sum_{n=1}^{\infty} \frac{x^n}{n!}, \quad -\infty < x < \infty$$

Позначки Σ та Π без індексів просто записувались у двійкових командах МЕСМ. Із документацією по МЕСМ були надані приклади, як із використанням машинних команд записувати послідовне звернення до значень елементів векторів та матриць в циклічно повторюваних діях.

Необхідність провести однакові розрахунки для кожного значення з певної множини даних позначались квантором загальності \forall . Це визначало потребу у повторенні дій для кожного елементу з множини значень, тобто повторювати дії на кшталт циклування. Саме ці потреби привели до використання опосередкованої адресації 2-ого рангу (вказівників), що дозволило економити час та зменшити ймовірність помилок під час введення даних.

А адресація 2-ого рангу до адрес підпрограм дозволила підвищити універсальність програм та підпрограм [43]. Вишукані прийоми використання опосередкованої адресації 2-ого рангу, які винайшли кияни, не описано в документації по МЕСМ та відсутні в операторному програмуванні. Подальше узагальнення опосередкованої адресації на адресацію вищих рангів дозволило з використанням «штрих-операції» (розіменування вказівників) формалізувати відношення слідування комірок пам'яті, які розташовані в пам'яті комп'ютера за довільними адресами.

Автоматизації програмування спрямовувались на полегшення роботи програмістів. Мнемонічні коди намагалися наблизити до інструкцій, які складали для розрахунків арифмометрами, що викликало набуття мнемонічними кодами ознак високорівневого програмування.

У програмуванні на МЕСМ також використовувався відомий бібліотечний метод програмування [11, 21, 22]. Для МЕСМ була розроблена бібліотека стандартних підпрограм з підпрограми елементарних функцій.

Інженери, які розробили МЕСМ, не збиралися на цьому зупинитися, до їхніх планів входила розробка великого київського комп'ютера, архітектуру якого розробили у 1952-1954р. Офіційно почали збирати комп'ютер у 1954р, який згодом у 1956р. отримав назву «Київ». Досвід використання МЕСМ та роботи з автоматизації програмування підказали математикам та інженерам важливі та унікальні архітектурні рішення для асинхронного комп'ютера «Київ» широкого призначення.

8. Високорівневі машинні програми комп'ютера «Київ»

«Київ» та Адресна мова. Ученій Катерині Ющенко доручили розробку математичного забезпечення комп'ютера «Київ», включаючи розробку системи команд, системи кодування даних у пам'яті, бібліотечних підпрограм та інші питання, пов'язані з програмуванням.

Процес написання програм у машинних кодах для МЕСМ був трудомістким та незручним, оскільки команди для виконання процесором записувались у двійковому коді, незручному для сприйняття людиною.

Під час розробки комп'ютера «Київ» враховувалися потреби полегшення роботи не лише програмістам, а й обчислювачам. К.О. Шкабара та К.Л. Ющенко ще задовго до початку робіт з проектування комп'ютера «Київ», обговорювали потребу та можливість апаратної реалізації вдосконалених мнемонічних кодів програм, можливості модифікації адрес для циклів. Проводилися дослідження заміни введення даних по окремим бітам на введення символів⁴.

Досвід розробки дослідницького компілятора виразів для МЕСМ та мова програмування, яка розроблювалась, вплинули на систему команд комп'ютера «Київ» [9 (Т. 1 стор. 30, 43)]. Система команд комп'ютера широкого призначення мала враховувати потреби розробки програмуючої програми (компілятора), можливості обробки текстів та бути зручною для розв'язку логічних та інших задач.

На відміну від асинхронного комп'ютера «Київ» широкого призначення, комп'ютери, розроблені до 1955р., призначалися виключно для математичних розрахунків [11, 26] шляхом виконання програм у двійкових машинних кодах або дуже до них наближених. Наприклад, до пам'яті комп'ютера ЕДСАК програми на асемблері вводились спеціальним пристроєм без задіяння процесора.

У комп'ютері «Київ» уперше в світі була апаратно реалізована адресація 2-ого рангу та можливість задання циклів [11 (стор. 489)].

Відтак комп'ютер широкого призначення «Київ» був першим у світі комп'ютером, у системі операцій якого було закладено потужні засоби програмування, які дозволяли визначати та обробляти спискові структури.

Оскільки операції комп'ютера «Київ» були ближче до мнемонічних кодів, ніж команди МЕСМ, то ті ж програми у мнемонічних кодах легше вводились до пам'яті комп'ютера «Київ» ніж до МЕСМ. Згодом, як зазначається в анотації до [40], Адресна мова набуває призначення для перенесення програм між комп'ютерами з різною архітектурою.

Автоматизацію програмування згодом називають адресним методом програмування, а сам термін «мова програмування» кияни починають застосовувати лише з 1958р. Деякий час вчені вважали, що Адресна мова створена у момент початку використання терміну «мова програмування» для адресного методу програмування. У ті роки ОЦ АН УРСР набував обертів та для більшої вагомості його досягнень було оголошено, що Адресну мову запропоновано в його стінах. Згодом було визначено, що Адресна мова створена 1955р., саме тоді, коли було запропоновано адресний метод програмування та запущено процесор комп'ютера «Київ» з апаратно реалізованою головною, базовою складовою Адресної мови програмування [11 (Т. 1 стор. 61)].

Згідно зі спогадами інженерів та очевидців С.Б. Погребинського і Б.М. Ма-

⁴ В комп'ютері «Київ» символи кодуються трьома вісімковими знаками.

линовського комп'ютер «Київ» був розроблений 1956р [18]. Під час завершальної стадії розробки до комп'ютера «Київ» приєднувались унікальні зовнішні пристрої введення та виведення даних. Після перевезення Комп'ютера «Київ» з Феофанії на вул. Лисогірську 14.03.1958р. відбувся його повторний запуск. Порівняно з МЕСМ «Київ» мав не лише значно простіше програмування, а й мав продуктивність у сотні разів більшу ніж МЕСМ. Програми, які виконувались на МЕСМ декілька годин «Київ» виконував за лічені хвилини. 1956р. було прийнято рішення передати МЕСМ до КПІ [15, 18] для навчання студентів. Восени 1959р. було завершено перевезення МЕСМ з Феофанії до КПІ.

Команда складання команд. Широкої можливості команди складання команд МЕСМ та, водночас, незручність її використання, визначили напрями її вдосконалення та узагальнення. Також стало зрозуміло, що для прискорення динамічної модифікації адрес, необхідно мати спеціалізовані комірки у складі самого процесора, тобто реєстри. Реєстри модифікації адрес виконують більше функцій, аніж загальновідомі адресні реєстри. Реєстри модифікації адрес призначались, окрім прискорення доступу за індексами до елементів масивів, ще й для опосередкованої адресації 2-ого рангу – «штрих-операції» (розіменування вказівників) [2, 11 (Т. 1 стор. 489), 10].

Розробка системи команд комп'ютера «Київ» вдосконалила можливості динамічної зміни програм МЕСМ для забезпечення можливості та зручності розробки програмуючих програм (компіляторів). Це було революційним рішенням та видатним винаходом у галузі ІТ. Динамічну зміну програм комп'ютера «Київ» забезпечують групові операції з використанням реєстру модифікації адрес. До групових операцій належать три операції, включаючи Ф-операцію, яка заповнювала значенням реєстр модифікації адрес [2-5, 7, 11, 10] та ще дві групові операції.

В окремому розділі глави III в монографії [2, 10] детально описано групові операції комп'ютера «Київ» та зокрема

Ф-операція. З цього розділу видно, що, окрім розіменування вказівників, система операцій комп'ютера «Київ» передбачає задання даних, організованих у спискові ланцюжки, ідентичні найпростішому типу абстрактних типів даних. Також легко побачити, що Ф-операція дозволяє визначати та обробляти дані, які групуються у спискові ланцюжки, ідентичні абстрактним типам даних – однозв'язним лінійними списками та іншим складним ієрархічним структурами. На відміну від абстрактних типів даних, елементами цих структур можуть бути окрім даних і підпрограми, аналогічно ООП.

Групові операції визначають початок та кінець циклування та фактично являють собою заголовки циклів імперативних мов програмування або, як визначено у Т. 1 на стор. 489 в [11], – «*операції для задавання циклів*». Задання циклів операціями в комп'ютері «Київ» має унікальну можливість – **задавати цикли за елементами «списків»**.

Слід зазначити, що Ф-операція уможливило визначення одного з двох типів відношень слідування:

1) послідовне слідування комірок оперативної пам'яті (для послідовного звернення до елементів списків та простих структур);

2) слідування, яке визначається послідовними значеннями вказівників («штрих-операцією»), які визначають «список» (для послідовного звернення до елементів «списків»).

Композиція цих двох відношень слідування в Адресній мові використовується для визначення довільних складних ієрархічних структур. Елементами, які належать послідовним слідуванням комірок, можуть бути адреси, які «породжують» нові «адресні» ланцюжки, що утворюються застосуванням «штрих-операції». Ідентично в Асоціативному програмуванні будуються складні спискові структури: «*у структуру спискового члену можуть входити описи списків*» [20 (стор. 110)].

У комп'ютері «Київ» є команда з назвою «команда складання команд», яка ідентична однойменній команді МЕСМ. Вона може бути використана для вико-

нання «штрих-операції» (розіменування Pointers).

Окреме, принципове значення має можливість використання Ф-операції з операцією завершення групових операцій для виконання багатократного виконання «штрих-операції», тобто для багатократного розіменування вказівників (Multiple indirection of Pointers).

Таким чином операції групові операції дозволяють задавати цикли зі змінною циклу та/або по елементам масивів (у загальному випадку по елементам багатовимірних масивів), а Ф-операція з ЗГО – цикли по елементам «списків». Групові операції являють собою апаратну реалізацію обробки складних ієрархічних структур (деревоподібних форматів) та призначені для прискорення їх обробки [15, 18].

В тілі циклу можуть використовуватися команди умовного чи безумовного переходу для «дострокового» виходу з циклу.

Високорівневість операцій. Відтак, у системі команд комп'ютера «Київ» наявні можливості організації циклічних процесів з послідовної обробки змінних числових значень, елементів масивів і елементів «списків».

До унікальних можливостей комп'ютера «Київ» належить умовна передача управління за опосередкованою адресою вищих рангів та відносна передача управління, яка дозволила завантажувати програми без будь-яких її змін у довільну ділянку оперативної пам'яті так, щоб робота програми не залежала від її місця розташування в оперативній пам'яті комп'ютера [2, 11 (Т. 1 стор. 61) 12, 10].

Система команд комп'ютера «Київ» містить, окрім групових операції модифікації адрес (засоби організації циклів), умовні та безумовні переходи, розгалуження можливості організувати цикли з перевіркою умов завершення виконання циклів. Команди вхідної мови програмування комп'ютера «Київ» мають оператори засилення (присвоєння) та можливість отримувати значення безпосередньо за адресою та за опосередкованою адресою 2-ого (адреса адреси або вказівник) та більш високого рангу. Вхідна мова комп'ютера «Київ» має

засоби звернення до елемента масиву за його порядковим номером (індексом).

Відтак, вхідна мнемонічна мова комп'ютера «Київ» з можливістю задавати цикли та з використанням опосередкованої адресації визначати та оброблювати складні спискові структури (деревоподібні формати) [11 (Т. 1 стор. 489)] являє собою мову програмування високого рівня, оскільки має більше можливостей ніж мова «Планкалькюль» Конрада Цузе, яка визнана першою мовою програмування високого рівня.

Українські вчені мають усі підстави для визнання міжнародною спільнотою першості України у винаході як самих вказівників, так і у винаході списків та складних структур – деревоподібних форматів [20, 26, 40]. Абстрактні типи даних являють собою частковий випадок деревоподібних форматів, оскільки в деревоподібних форматах допускаються циклічні посилання на довільні комірки оперативної пам'яті, а як їхні елементи можуть використовуватися, подібно до методів ООП, підпрограми.

У комп'ютері «Київ» було вперше застосовано технологію використання змінно спаяних модулів з бібліотеками підпрограм, що дозволило суттєво економити оперативну пам'ять комп'ютера при використанні підпрограм.

Обробка чисел із плаваючою точкою була реалізована програмно. Прийняття такого рішення дозволило спростити інженерну компоненту комп'ютера «Київ», зменшити вартість та прискорити його розробку. Ці фактори переважили швидкість розрахунків з плаваючою точністю, оскільки було зрозуміло, що «Київ» проводитиме обчислення значно швидше за МЕСМ.

9. Унікальність концепції

Адресного програмування

У цьому розділі на прикладах показано, що циклування в Адресній мові є більш загальним та, на відміну від імперативних мов програмування, припускає визначення в заголовках циклу перегляд елементів «списків». Також у цьому розділі визначено окремі унікальні можливості Адресного програмування щодо обробки

даних, які містяться в складних ланцюжкових ієрархічних структурах – деревоподібних форматах.

Необхідно нагадати, що концепція деревоподібних форматів базується на двох типах відношення слідування комірок пам'яті. Перший тип відношення слідування визначається лінійним порядком слідування адрес в оперативній пам'яті. Другий тип адресного відношення слідування задається «штрих-операцією» та визначає спискові ланцюжки. Як композиція двох типів слідування дозволяє визначати довільні ієрархічні структури визначалось у попередньому розділі. Зауважимо, що елементами деревоподібних форматів можуть бути, окрім самих даних, і адреси підпрограм (подібно ООП).

Нагадаємо, що операції комп'ютера «Київ» дозволяють задавати цикли поза обома відношеннями слідування.

Адресна мова програмування передбачає задання циклів не лише зі змінними циклу арифметичного типу, а й за адресами (тип Pointer). Можна визначати перший елемент, із яким має виконуватись перший раз тіло циклу, наприклад «головою списку». Як крок (STEP) у циклах типу «FOR і FROM 1 STEP 1 TO N» можна вказати адресне слідування – «штрих-операцію» (розіменування вказівників). В імперативних мовах з вказівниками необхідно використати Until або While. В табл. 1 наведено приклад фрагментів мовою C++ та Адресною мовою, в яких циклічно виконуються дії F із кожним елементом масиву та списку. Адресний тип відношення слідування (з використанням «штрих-операції») надає Адресній мові програмування можливість оперувати поняттям «множина».

Літери Π та Π – це позначки формули циклування та формули входження (виклику) підпрограм. Літера π є аналогом змінної циклу i та змінної **Current**, значення яких зберігається у регістрах. Запису $\Pi\{ List, ('\emptyset) \Rightarrow \pi \}$ відповідають групові операції комп'ютера «Київ». Групові операції та використання регістру, а не комірок оперативної пам'яті, забезпечують прискорену обробки складних структур [2, 11, 10, 40, 41].

В Адресній мові програмістам надається можливість оперувати поняттям «множина» у визначенні повторювальних дій з кожним елементом множини або для тих елементів множини, які задовольняють визначену програмістом умову. Циклування в Адресній мові дозволяє задавати цикли по множинам, які є об'єднанням інших множин. Крім сказаного, велике значення для зручності програмістів має можливість визначати однією формулою циклування («заголовком циклу») одразу і вкладені цикли. Для задання циклу програмісту достатньо визначити множину та один із типів відношення слідування на ній.

У формулі циклування можна використовувати конструкції фільтрування, аналогічні умовам Where мови SQL.

Функція «map», яка наразі широко використовується у мовах програмування, природно визначається формулою циклування в Адресній мові.

Широкі можливості визначення відношення адресного слідування у формулах циклування дозволяють реалізовувати різноманітні схеми огляду областей доступності та деревоподібних форматів, що відповідає сучасному терміну «спосіб об-

Таблиця 1

Приклади фрагментів програм із циклом за елементами масиву та елементами списку

| | C++ | Адресна мова | Пояснення на мові, подібній до C++ |
|---------------------------|--|--|---|
| Цикл за елементами масиву | <pre>for int i (0; n, ++i) { F(M[i]); }</pre> | $\Pi\{ 0, (\emptyset \oplus 1, n) \Rightarrow \pi \}$ $\Pi F\{\pi\}$ | – |
| Цикл за елементами списку | <pre>List *Current = Head; while (Current -> Next != Null) { F(Current->info); Current = Current -> Next; }</pre> | $\Pi\{ List, ('\emptyset) \Rightarrow \pi \}$ $\Pi F\{\pi \oplus 1\}$ | // подібного в сучасних мовах немає: for *Pi (List; Null, *List) { F(Pi); } |

ходу дерев» в ширину, глибину, праворуч чи ліворуч.

При визначенні циклування програмісту надається можливість користуватися поняттями «множина» та «підмножина».

В сучасних імперативних мовах аналогу «мінус штрих-операції», яка є оберненою до «штрих-операції» (розіменування Pointers) не існує. Ця операція має суто декларативний характер та дозволяє «пробігати» елементи однозв'язних лінійних списків у зворотному порядку [39, 40, 45].

Окрім того, «мінус штрих-операція» дозволяє визначати дерева шляхом визначення лише батька для вузлів дерева. В імперативних мовах із використанням концепції абстрактних типів даних для вузлів необхідно визначити список синів, наприклад: «лівий син», «правий син», «середній син» [45].

В Адресному програмуванні визначено поняття багатовимірного адресного сортування, яке дозволяє мати одночасно декілька різних варіантів сортування елементів множини. При адресному сортуванні відсутня необхідність змінювати місцезнаходження елементів множини під час їх сортування [25, 45].

10. Використання та значення Адресної мови програмування

Адресна мова програмування вплинула на вибір операцій не лише для комп'ютера «Київ», а й для інших київських комп'ютерів «Дніпро», «Промінь», «Дніпро-2», «Мір» [9 (Т. 1 стор. 30), 35 (стор. 10), 31]. Ідеї киян щодо модифікації адрес, були враховані в архітектурі інших радянських комп'ютерів, зокрема в «М-20» і архітектурно сумісними з нею: БЕСМ-3, БЕСМ-3М та БЕСМ-4 [8], в комп'ютерах серій «Мінськ», «Урал» та інших.

У відео [6 з 1:31:52] доцент факультету прикладних наук Українського Католицького Університету Олег Фаренюк, фахівець у галузі архітектури комп'ютерів, висунув та обґрунтував тезу: *«можна вважати, що «Київ» реалізував ідею RISC за кілька десятиліть до появи цього терміну.»*

Адресна мова використовувалась у розробці унікальних систем, зокрема, для реалізації інформаційної системи «Авто-

директор» – табличної бази даних реляційного типу [18].

В монографії по комп'ютеру «Київ» [2, 10 (стор. 97-110)] наведено перелік реалізованих програм та систем. Це система автоматизованого проектування електричних плат, декілька компіляторів, емулятор архітектури комп'ютера «Дніпро», інші розв'язки методами навчання задач розпізнавання образів, серед них, простих геометричних фігур, друкованих рукописних букв і чисел. І це не повний перелік реалізованих комп'ютером «Київ» задач.

Слід зазначити, що Адресна мова передбачає паралельне виконання операторів [2-5, 10, 40-42]. Можливість паралельного виконання програм Адресною мовою була використана для реалізації паралельних асинхронних процесів комп'ютера «Київ».

У період з 1956р. до кінця 1959р. в Адресній мові були конкретизовані та уточнені поняття слідування адрес, області доступності, введено поняття деревоподібних форматів [26, 40, 45]. Досліджувались методи та розроблювались схеми огляду даних, областей доступності або деревоподібних форматів. Надзвичайно важливого подальшого розвитку набули теоретичні дослідження деревоподібних форматів, які мали практичне значення для застосування універсальних засобів групування даних та процедур їх обробки [40].

До одного з найцікавіших та найпотужніших засобів розвитку Адресної мови програмування належить винахід «мінус штрих-операції», яка є оберненою до «штрих-операції» (розіменування вказівників) [11, 39, 40, 45]. Ця операція була реалізована В.П. Сьомиком у компіляторах Адресної мови [39]. «Мінус штрих-операція» носить виключно декларативний характер та тому не має аналогів у сучасних імперативних мовах програмування. В [45] описано порівняння можливостей «мінус штрих-операції» з можливостями сучасних декларативних засобів програмування та зроблено висновок, що Адресна мова програмування має вичерпні декларативні можливості.

Для комп'ютера «Київ» були розроблені унікальні пристрої введення та виведення даних. До цих пристроїв належать

пристрої введення графічних зображень з паперових носіїв інформації, фотоплівок і фотопластин [18] та пристрої друку зображень на папері [18].

Інформація в пам'яті комп'ютера про зображення зберігалась одним з двох способів: шляхом представлення по точкам або аналітичними функціями. На Адресною мовою були розроблені бібліотеки програм для обробки зображень, зокрема, розв'язувались задачі перетворення точкового представлення зображення у аналітичну форму. Проблеми перетворення точкового представлення зображень у аналітичне спонукали видатного математика В.Л. Рвачова до винайдення R-функцій, які згодом *«були успішно використані в задачах розпізнавання образів»* [34 (стор. 50)]. Це були значні національні досягнення, що перевершували тогочасні закордонні винаходи. Наявність пристроїв введення зображень в пам'ять комп'ютера та потужні можливості опосередкованої адресації вищих рангів дозволили українським вченим розробити систему розпізнавання простих геометричних фігур, друкованих літер та цифр [2, 7, 10, 18, 34].

Для комп'ютера «Київ» був розроблений пристрій отримання та передачі даних через телеграфні лінії зв'язку. Цю можливість було використано для віддаленого керування технологічними процесами на сталеплавильному заводі в місті Кам'янське (колишній Дніпродзержинськ), який знаходився від комп'ютера «Київ» на відстані 500 км. [18].

Другий примірник комп'ютера «Київ» було зібрано у 1959р. на замовлення Об'єднаного інституту ядерних досліджень (м. Дубна), де він понад десять років успішно використовувався для розв'язку задач [18, 30].

На комп'ютері «Київ» було створено емулятор комп'ютера «Дніпро». На цьому емуляторі розроблено компілятор Адресної мови для комп'ютера «Дніпро» [2, 10]. Це мало велике значення, оскільки дозволило ще до моменту запуску цієї машини, мати готовий компілятор Адресної мови та низку інших програм [2, 11, 10].

Це унікальний випадок в історії зародження інформаційних технологій, коли написання та налагодження програм здійснювалось для ще неіснуючого комп'ютера «Дніпро».

На комп'ютері «Київ» проводилися роботи із автоматизованого проектування електричних схем вузлів для комп'ютерів, що знаходилися на стадії розробки. Це давало можливість віртуально тестувати електричні схеми без їх фактичного виготовлення. З використанням комп'ютера «Київ» та Адресної мови програмування розроблено комп'ютери «Дніпро», Промінь, комп'ютери серії «МІР» та комп'ютер «Дніпро-2» [11, 31].

Адресною мовою програмування було написано низку синтаксичних аналізаторів та розроблено цілий ряд компіляторів з інших мов програмування [11, 12, 33].

Наведемо цитату [11 (Т. 1 стор. 63)]: *«Завдяки можливості описувати адреси як функції якихось параметрів Адресної мови можна описувати й довільні схеми огляду інформації та складні інформаційно-логічні й економічні алгоритми і складні процеси перегляду й пошуку інформації, організованої в ланцюгові списки і спискові структури; алгоритм, процеси такого роду не можна описувати за допомогою алгоритм, мов типу АЛГОЛ, не залучаючи допоміжних засобів. У цьому відношенні Адресна мова випередила алгоритмічні мови, створені за кордоном для спискової обробки символічних виразів (напр., ЛІСП тощо).»*

Адресна мова належить до мов функційного програмування («мов спискових») ⁵ [11 (Т. 2 стор. 86, 87), 20 (стор. 107)]. Вона запропонована на декілька років раніше за мову LISP, яку помилково вважають першою мовою функційного програмування.

Адресна мова програмування реалізована на багатьох радянських комп'ютерах 1-ого та 2-ого покоління, зокрема, на комп'ютерах «Київ», М-20 та сумісних з ним БЕСМ-3, БЕСМ-3М та БЕСМ-4, комп'ютерах серій «Урал» та «Мінськ» та

⁵ У 50-х та 60-х роках замість «функціональні мови» використовувався термін «мови спискові».

використовувалася програмістами пост-соціалістичного простору понад 20 років [18, 33]. Концепції рангу адрес та деревоподібних форматів увійшли до багатьох радянських мов програмування, зокрема, до: АЛГЕМ, А-КОБОЛ [11 (Т. 1 стор. 63)] та А-АЛГАМС [28].

Машинно-орієнтовані мови програмування АВТОКОД, зокрема для комп'ютера «Дніпро-2» [27, 28], включають засоби опосередкованої адресації вищих рангів Адресної мови програмування, що дозволяє визначати та оброблювати складні ієрархічні структури даних.

Видатний учений А.І. Кітов, відомий теорією Асоціативного програмування (1963р.) та мовою АЛГЕМ, в [20 (стор. 113)] зазначив, що принцип рангу адреси та адресних функцій Адресної мови програмування використано в Асоціативному програмуванні та мові АЛГЕМ, де складні ієрархічні структури утворюються використанням двох типів членів списків: вузлових та гніздових. Ці типи ідентичні двом типам відношень слідування Адресного програмування. У термінології Асоціативного програмування [20] збережено поняття «адреса» Адресної концепції програмування, а поняття «адресне співвідношення» ідентичне формулі засилання та відповідає адресному відображенню.

Система програмування Альфамовою (1963р.) для комп'ютера М-20 (та сумісних з ним: БЕСМ-3, БЕСМ-3М та БЕСМ-4), яка розроблена у середині 60-х років, містить засоби, які термінологічно, семантично та частково синтаксично ідентичні засобам Адресної мови програмування. Це засоби визначення циклів та спискових структур розміщення інформації. Система АЛГІБР є модифікацією Альфа-мови для БЕСМ-6 [11 (Т. 1 стор. 102-104), 16, 17].

Висновки

У статті розглянуто поступове зародження Адресної мови програмування з опосередкованою адресацією вищих рангів та складними ієрархічними структурами: деревоподібними форматами. Визначено вплив унікальної архітектури МЕСМ на винахід Адресної мови.

Стаття відновлює забуту та втрачену сторінку виникнення програмування високого рівня в Україні.

Світова спільнота ще не визнала першість винаходу вказівників за Україною та продовжує помилково вважати, що перші вказівники винайдені Гарольдом Лоусоном 1964р.

В статті наведено та детально розглянуто наявні публікації щодо підтверджень винаходу українськими вченими потужних засобів програмування: опосередкованої адресації (вказівників) вищих рангів та деревоподібних форматів, до яких подібні абстрактні типи даних. До переліку підтверджень належить низка публікацій з Адресної мови програмування та архітектури комп'ютера «Київ», перекладених багатьма мовами та виданих за кордоном.

Результати статті стануть в пригоді для визнання світовою спільнотою першості за Україною винаходу вказівників та складних ієрархічних структур у програмуванні високого рівня.

Стаття має за мету популяризацію досягнень українських науковців у часи зародження інформаційних технологій та має важливе значення для національно-патріотичного виховання молоді, оскільки спонукає молодь до наукових досліджень.

Стаття має значення для збереження всесвітньої історії зародження та розвитку інформаційних технологій, зокрема історії зародження програмування високого рівня з вказівниками та універсальними потужними засобами групування і зв'язування даних та процедур у складні спискові ієрархічні структури.

Література

1. Alvaro Videla, M 2018, 'Kateryna L. Yushchenko — Inventor of Pointers' A Computer of One's Own – Pioneers of the Computing Age, blog post, 8 December, viewed 24 October 2021, <https://medium.com/a-computer-of-ones-own/kateryna-l-yushchenko-inventor-of-pointers-6f2796fa1798?fbclid=IwAR3fcqmC0COfy5EqyIHBrIqHcPno5MUFZjCUQ-SM-vxhD0g3xbj_P2SRCM>.
2. Glushkov V.M., & Yushchenko E.L., D 1966, The Kiev Computer; a Mathematical Description, USA, Ohio, Translation Division,

- Foreign Technology Div., Wright-Patterson AFB, 234p., ASIN: B0007G3QGC.
3. Gnedenko B.V., Koroliouk V. S. & Iouchtchenko E.L., D 1969, *Éléments de programmation sur ordinateurs*, Paris, Dunod, 362p., ASIN: B0014UQTU0, viewed 24 October 2021, <https://files.infoua.net/yushchenko/Elements-de-programmation-sur-ordinateurs_BGnedenko-VKoroliouk-EIouchtchenko_1969_France_OCR.pdf>.
 4. Gnedenko B.V., Koroljuk V.S. & Justschenko E.L., D 1964, *Elemente der Programmierung*, DDR, Leipzig, Verlag: B. G. Teubner, 327 oldal.
 5. Gnedenko B.V., Korolyuk V.S. & Juscsenko E.L. D 1964, *Bevezetés a progamozásba*, – I, II. – Magyarország, Budapest, Uj technica.
 6. Sensei Yuri. Унікальність комп'ютера «Київ», 2021. YouTube. URL: <https://youtu.be/3ohE3njX8P0?t=5512> (дата звернення: 17.11.2021).
 7. Ware Willis H. & Holland Wade B., D 1963, *Willis H. Ware. Soviet Cybernetics Technology: III, Programming Elements of the BESM, STRELA, Ural, M-3, and Kiev Computers*, Translated by A.S. Kozak, RAND Corporation, 91p., Series: Research Memoranda, USA, viewed 24 October 2021, 70p., viewed 24 October 2021, <https://files.infoua.net/yushchenko/Soviet-Cybernetics-Technology-III_1963.pdf>.
 8. Вакуленко С. Вычислительная машина «Киев», математическое описание. Dreamwidth.org. URL: <https://vak.dreamwidth.org/771441.html> (дата звернення: 31.10.2021).
 9. Васильев Ю. Его оружие — математика К 60-летию Алексея Андреевича Ляпунова. / в кн.: Алексей Андреевич Ляпунов / Редакторы-составители Н.А. Ляпунова, Я.И. Фет. – Новосибирск : Филиал «Гео» Изд. СО РАН, Издательство ИВМиМГ СО РАН, 2001. – 502 с., – URL: www.ict.nsc.ru/jspui/bitstream/ICT/1260/1/Lyap_2001.pdf (дата звернення: 11.10.2021).
 10. Глушков В.М., Вычислительная машина «Киев». Математическое описание / В.М. Глушков, Е.Л. Ющенко. // К. : – Гостехиздат УССР, 1962. – 183 с. : ил., – URL: [Vychislitel'naya-mashyna-Kiev_VHlushkov_EYushchenko_1962.pdf](https://files.infoua.net/VHlushkov_EYushchenko_1962.pdf) (infoua.net) (дата звернення: 21.10.2021).
 11. Глушков В.М. (відпов. ред.), *Энциклопедия кибернетики*. К. : Головна ред. укр. рад. енциклопедії, 1973. Т. 1 : А-Л. 584 с., Т. 2 : М - Я. 574 с. URL: https://files.infoua.net/yushchenko/Encyklopediya-kibernetyky_T1-2_VHlushkov-ta-in_1973_UKR_DJVU.zip (дата звернення: 24.10.2021).
 12. Глушков В.М. и др., *Основные направления развития цифровой вычислительной техники : обзор* / В.М. Глушков, Б.Н. Малиновский, З.Л. Рабинович, Е.Л. Ющенко. – М. : – ЦНИИТЭИ приборостроения, 1967. – 96 с.
 13. Гнеденко Б.В. Воспоминания. Моя жизнь в математике и математика в моей жизни. / Б.В. Гнеденко. // под ред. Д.Б. Гнеденко / (Наука в СССР. Через тернии к звездам), – М. : Ленанд, 2015. – 624 с., ISBN 978-5-9710-1416-4.
 14. Гнеденко Б.В., *Элементы программирования* / Гнеденко Б.В., Королюк В.С., Ющенко Е.Л.; // М. : – ГИФМЛ, 1961. – 348 с., – URL: [Elementy-programmirovaniya_BGnedenko-VKoroljuk-EYushchenko_1961.pdf](https://files.infoua.net/BGnedenko-VKoroljuk-EYushchenko_1961.pdf) (infoua.net) (дата звернення: 23.10.2021).
 15. Дашевский Л.Н., Шкабара Е.А., Как это начиналось. // Новое в жизни, науке, технике. Сер. Математика, кибернетика; № 1 / М. : Знание, 1981. – 64 с., – URL: [kak_eto_nachinalos.pdf](http://computer-museum.ru/kak_eto_nachinalos.pdf) (computer-museum.ru) (дата звернення: 23.10.2021).
 16. Ершов А. П., *Руководство к пользованию системой Альфа*. / Ершов А. П., Кожухин Г.И., Поттосин И.В. // Новосибирск : Наука, 1968. – 180 с.
 17. Ершов А.П., *Входной язык системы автоматического программирования*. // Ершов А.П., Кожухин Г.И., Волошин Ю.М. / М. : 1961. – С.173-174.
 18. *Європейський віртуальний музей історії інформаційних технологій в Україні.*, – URL: <http://www.icfest.kiev.ua/MUSEUM/> (дата звернення: 07.10.2021).
 19. Іваненко Л.М. МЭСМ та її люди на відстані літ С.29-35 // В кн.: Видатні конструктори України. За матеріалами наукових читань з циклу ”Видатні конструктори України”, проведених у 2001–2018 роках. Т. 8. / кер. гр. уклад. М.Ю. Ільченко; за редакцією Б.Є. Патона, М.З. Згуровського. – К. : ВПК “Політехніка”, 2018. – 256 с., – URL: <http://biography.nbuv.gov.ua/rating/r2018/txt/g3/1007.pdf> (дата звернення: 23.10.2021).

20. Китов А.И. Программирование информационно-логических задач. М. : Советское радио, 1967. 124 с., – URL: <http://library.dnu.dp.ua/0318.rtf> (дата звернення: 26.10.2021).
21. Китов А.И., Криницкий Н. А., Комолов П.Н., Элементы программирования (для электронных цифровых машин). / Отв. ред. Китов А.И. М. : изд-во Артиллерийской академии им. Дзержинского. // М. : – 1956. – 288 с.
22. Китов В.А., Прохоров С.А. Становление программирования в СССР с 1950-го по 1960 год. Виртуальный компьютерный музей, Материалы международной конференции SORUCOM 2011. – URL: https://www.computer-museum.ru/histsoft/1950_1960_sorucom_2011.htm (дата звернення: 29.09.2021).
23. Коваленко І. М. Гранічні задачі життя. До 90-річчя академіка НАНУ В.С. Королюка / І.М. Коваленко, О.А. Летичевський, І.В. Сергієнко // ISSN 1027-3239. Вісн. НАНУ, № 8, 2015. С.95-99, – URL: <http://www.visnyk-nanu.org.ua/sites/default/files/files/Visn.2015/8/15.Kovalenko.pdf> (дата звернення: 26.10.2021).
24. Кратко М. «3 історії розвитку інформатики в Україні: [історія створення електронних обчислювальних машин]», Світогляд. № 6, 2009. – С.56-62, – URL: https://www.mao.kiev.ua/biblio/jscans/svitogliad/svit-2009-20-6/svit-2009-20-6-56-kratko.pdf?fbclid=IwAR2mHMXkg_cVlKzi4Pywb2e-dGLXEk9IdW0pX6q1Yy7uyDkGB35eMAGY (дата звернення 21.10.2021).
25. Крещенко Т.О. Метод кластеризації з використанням багатовимірного адресного сортування. / Крещенко Т.О., Ющенко Ю.О. // Наукові записки НаУКМА. – Т. 3 : Комп'ютерні науки. – К. : – 2020. – С.83–87, – URL: <https://doi.org/10.18523/2617-3808.2020.3.83-87> (дата звернення 12.10.2021).
26. Кулинкович А.Е., Ющенко Е.Л., О базовом алгоритмическом языке. / Кулинкович А.Е., Ющенко Е.Л., в журн.: «Кибернетика», К. : № 2, 1965. С.3-9, – URL: https://files.infoua.net/yushchenko/O-bazovom-algoritmicheskoy-yazyke_AKulinkovich_EYushchenko_1965.pdf (дата звернення 12.10.2021).
27. Лаврищева Е.М. АКД – Автокод машины «ДНЕПР-2». / Лаврищева Е.М., Никитин А.И., Усенко Л.Г., Ющенко Е.Л. // К. : Ин-т кибернетики АН УССР, 1969. – 97 с.
28. Лаврищева Е.М., Л.Г.Борисенко, Усенко Л.Г., Ющенко Е.Л. Транслятор Д-АЛГАМС для УВК Днепр-2. / Лаврищева Е.М., Борисенко Л.Г., Усенко Л.Г., Ющенко Е.Л. // К. : ИК АН УССР, 1971. – 246 с.
29. Лебедев С.А., Дашевский Л.Н., Шкабара Е.Л., Малая электронная счетная машина. // М. : – Из-во АН СССР, 1952. – 162 с.
30. Малиновський Б.М., Історія обчислювальної техніки в обличчях. // К. : Фірма «КІТ», ПТОО «А.С.К.» 1995. 245 с., – URL: http://icfst.kiev.ua/MUSEUM/TXT/Malinovsky_history_ukr.pdf (дата звернення 21.10.2021).
31. Міщенко Н., Інститут кібернетики НАНУ: все починалося у Феюфанії... (1956–1958) : спогади. Ч. 1 / Надія Міщенко // Кібернетика, інформатика і довкола... : наукові праці, повідомлення, спогади., – URL: <https://cyberua.info/novyny/ik-nanu-vse-pochynalosja-u-feofaniji-sphohady-ch1-nadijamishchenko/> – (дата звернення: 26.10.2021).
32. Олешко А.Л., Коваленко (Юфіт) Нонна Наумовна (вспоминания) / Газета «Південна зоря». – Бердянська суспільно-політична газета. – № 1306, архів: С.295-344 – URL: <http://pivdenka.berdyansk.net/assets/files/book/Odnokursniki/odnokursniki-chast08.pdf?fbclid=IwAR3XkFHgPhLhmm-ScbTbS9cjs3LHthBENTp-JsFQmfZ3eSM-j4UsmeS9Ik> (дата звернення: 26.10.2021).
33. Перевозчикова О.Л., Школа теории программирования Е.Л. Ющенко. К : – Наука та наукознавство. – 2007. – № 4. – С.114-146, – URL: <http://dspace.nbuv.gov.ua/handle/123456789/49192?show=full> (дата звернення: 05.10.2021).
34. Подгаєцький О. Еволюція розробок у галузі штучного інтелекту в Україні та світі (Evolution of research in the field of artificial intelligence in Ukraine and the World) / Oleksandr Podgayetsky // Дослідження з історії техніки. – № 16, Харків : – 2012. – URL: <https://ela.kpi.ua/bitstream/123456789/7703/1/RHT-issue-16-title-05-Podgayetsky.pdf> (дата звернення: 07.10.2021).
35. Семик В.П. Транслятор с адресного языка для ЭВМ «Минск-1» / Семик В.П., Згурский А.Д., Шкляр Л.И., Глушко А.Г. // Сб. «Приборы и устройства средств авто-

- матики и телемеханики». Вып.1. Изд. ХГУ. – г.Харьков : 1965. С.96-98.
36. Семик В.П., Ющенко Е.Л. Адресный язык и принцип адресности в алгоритмических языках. // Сб. «Информационные системы». Изд. ВИНТИ АН СССР. – М. : 1964. С.9-14.
37. Семик В.П., Ющенко Е.Л. Возможности адресного языка для описания информационно-логических алгоритмов. // Сб. «Информационные системы». Изд. ВИНТИ АН СССР. – М. : 1964. С.5-8.
38. Сергієнко І.В., Інформатика в Україні: становлення, розвиток, проблеми / відп. ред.: Ю.В. Капітонова, Т.Т. Лебедева; рец.: Н.З. Шор, О.В. Палагін, К.Л. Ющенко; НАНУ, Ін-т кібернетики // – К. : Наукова думка, 1999. 354 с., – ISBN 966-00-0540-7.
39. Сьомик В.П., Про розширення поняття рангу адреси. // Сб. «Обчислювальна математика і техніка. Вид. АН УРСР. – К. : 1963. С.61-65.
40. Ющенко Е.Л., Адресное программирование // К. : – Гос. издательство технической литературы, УРСР, 1963. – 287 с., – URL: https://files.infoua.net/yushchenko/Adresnoe-programmirovaniye_EYushchenko_1963.pdf (дата звернення: 03.10.2021).
41. Ющенко Е.Л., Адресное программирование и особенности решения задач на машине «УРАЛ» / под ред. Бушко-Жук. // К. : Киев. высш. инженерное радиотехническое училище войск противовоздуш. обороны страны.: 1960. – 192 с.
42. Ющенко Е.Л., Адресный язык (Тема 5) // Кибернетика на транспорте: Заочный семинар. / Киевский дом Научно-технической пропаганды / – К. : – 1962. – 32 с., – URL: Kibernetika-na-transporte_Adresnyy-yazyk_KYushchenko_1962.pdf (infoua.net) (дата звернення: 12.10.2021).
43. Ющенко Ю.О., Pointers в програмах на «МЕСМ» / Для Європейського віртуального комп'ютерного музею історії інформаційних технологій в Україні: //К. : 2021, 9 с., – URL: http://www.icfest.kiev.ua/MUSEUM/TXT/YuriYushchenko_u.pdf (дата звернення: 29.09.2021).
44. Ющенко Ю.О., Катерина Логвинівна Ющенко – винахідниця Pointers та авторка однієї з перших в світі мов програмування високого рівня // Газета «Світ», № 5-6, 10.02.2021 р., Видав-во НАНУ та МОНУ. К. : – 2021. – С.2-3, – URL: https://www.nas.gov.ua/UA/Messages/Pages/View.aspx?MessageID=7487&fbclid=IwAR1BqiFUu7OGwN1knFbAUQt4sPeDKYZLkLeRIA BFxoX-RlgaxL_gtISJz-g (дата звернення: 23.10.2021).
45. Ющенко Ю.О., Окремі аспекти декларативності «мінус штрих-операції» // Наукові записки НаУКМА. – Т. 3 : Комп'ютерні науки. – К. : – 2020. – С.19–26, – URL: <http://nrpcmp.ukma.edu.ua/article/view/220657?fbclid=IwAR0N65fNM1liyasK84z00sEmzenb10FAy8lksOv0pzV6JSX54TJpBf0-V7o> (дата звернення: 03.10.2021).
46. Ядренко М.Й. Борис Володимирович Гнеденко – фундатор кафедри теорії імовірностей в Київському університеті. // Теорія імовірностей та математична статистика. Вип. 56, К : – 1997. – С.32-39, – URL: <https://probability.knu.ua/tims/issues-new/56/PDF/6.pdf> (дата звернення: 07.09.2021)

Одержано: 23.10.2021

Про автора:

Ющенко Юрій Олексійович,
к.ф.-м.н., доцент факультету інформатики
Національного університету
Києво-Могилянська Академія.
Кількість публікацій в українських
журналах – 16,
Індекс Хірша - 3.
ORCID: 0000-0003-4602-1774.

Місце роботи автора:

Національний університету
Києво-Могилянська Академія
Вул. Григорія Сковороди, 2, Київ,
Україна, 04655.
Електронна пошта:
yury.yuschenko@ukma.edu.ua.
Тел.: +38(093)3784051.