

УДК 372.8:004.43
EL 372.8:004.43

Лариса Іванівна КУБЛІЙ

кандидат технічних наук, доцент
кафедри автоматизації проектування
енергетичних процесів та систем,
НТУУ “Київський політехнічний
інститут імені Ігоря Сікорського”



e-mail: kublii_l_i@ukr.net

БАЗОВА МОВА ПРИ ВИВЧЕНІ СТУДЕНТАМИ АЛГОРИТМІЗАЦІЇ І ПРОГРАМУВАННЯ

Анотація. У статті розглянуто проблему вибору базової мови програмування при вивченні студентами дисципліни “Алгоритмізація та програмування” у першому семестрі першого року підготовки інженерів-програмістів. Проаналізовано використання мов Pascal і C як базових з врахуванням можливостей цих мов і рівня шкільної підготовки студентів-першокурсників.

Вказано особливості мови програмування C, які на початкових етапах відволікають студентів від опанування основ алгоритмізації і вироблення хорошого стилю програмування.

Аналіз можливостей спеціальних навчальних мов програмування і мови системного програмування C як першої навчальної показав, що використання спеціальної, вже класичної, навчальної мови Pascal (нову навчальну мову, яка б повніше відповідала сучасним вимогам, ще не розроблено) не потребує багато часу на її засвоєння,

сприяє зосередженню на алгоритмізації, від неї легко переходити до вивчення інших сучасних рейтинговіших мов програмування. Використання мови програмування С як базової на початкових етапах викликає ряд труднощів, пов'язаних з особливостями її синтаксису і відсутністю контролю помилок при виконанні програми, відволікає від алгоритмізації, і тому не є доцільним.

Ключові слова: *підготовка інженерів-програмістів, алгоритмізація, базова мова, мова програмування Паскаль, мова програмування С, структурне програмування.*

© КУБЛІЙ Л.І., 2017

Вступ. Від якості й результативності роботи програмістів залежить ефективність функціонування як конкретної фірми чи підприємства, так і рівень розвитку всіх сфер людської діяльності й країни в цілому. У зв'язку з цим потрібна якісна професійна підготовка майбутніх програмістів.

Щоб стати хорошим програмістом не досить вивчити якусь одну чи кілька найпопулярніших мов програмування. Насамперед треба навчитися програмувати — розробляти оптимальні алгоритми і коректно їх подавати. Для цього процес навчання має бути організований так, щоб зацікавити студентів, щоб вони прагнули більше навчитися, а головне, щоб з перших днів навчання їм було все зрозуміло. Важливу роль при цьому відіграє вибір базової мови програмування при вивченні дисципліни “Алгоритмізація та програмування”.

Протягом багатьох років на програмістській спеціальності, де я викладаю, першою мовою (лише перший семестр, щоб зрівняти знання і вміння тих, хто вивчав, до речі, на різному рівні, і не вивчав програмування в школі) була мова Pascal; у другому семестрі студенти

вивчали С і Асемблер. З цього навчального року вивчення алгоритмізації і програмування в першому семестрі за рішенням кафедри (ініціатива і абсолютна більшість була з боку молодих викладачів; основний аргумент — непопулярність мови Pascal) проводиться на базі мови С.

Аналіз останніх досліджень показав, що питання вибору мов програмування, з яких варто починати вивчення основ програмування в системі шкільної освіти залежно від віку учнів, розглядається в багатьох роботах науковців і вчителів. Питання ж вибору першої мови програмування для навчання основ програмування майбутніх інженерів-програмістів у вишах практично залишається поза увагою дослідників.

На важливість проблеми вибору мови програмування при навчанні вказано в роботі [1]: “Правильний вибір системи програмування, а отже, і алгоритмічної мови, відіграє чималу роль в ефективності майбутнього курсу інформатики”. Слушною також є позиція автора роботи [2]: “Поряд з базовими навичками алгоритміки й керування взаємодією програмних об’єктів важливим є вміння швидко опановувати нові мови і середовища програмування... Виправданим і бажаним є використовувати на різних етапах вивчення основ програмування різні середовища і навіть мови програмування”. У більшості випадків вибір мови програмування залежить від кваліфікації вчителів і від забезпеченості шкіл обчислювальною технікою. Як підсумок проведеної на сторінках журналу “Комп’ютер у школі та сім’ї” (номери 7 і 8 за 2013 [3, 4]) дискусії, де в основному на перше місце ставилися критерії доступності середовища програмування і його сумісність з сучасними операційними системами, В.В. Лапінський (V.V. Lapinskyi), Інститут педагогіки Національної академії педагогічних наук України, враховуючи думки переважної більшості

(близько 80%) фахівців, вважає, що за нинішніх реалій мовою навчання програмування може бути переважно Pascal, але не виключає можливості застосування інших мов, залишаючи право вибору за вчителем, залежно від його готовності до викладання [5]. В.А. Ребрина (V.A. Rebryna), Хмельницький обласний інститут післядипломної педагогічної освіти, вказує на низький рівень підготовки вчителів [2]: “Вхідне діагностування вчителів інформатики протягом останніх трьох років показало занадто низький рівень компетентностей вчителів з розв’язування задач як у консольному, так і у візуальному варіантах... Кваліфіковано вибирати мову програмування можуть не більше 25-30 відсотків”. Така ситуація негативно впливає на учнів — вони або взагалі втрачають інтерес до програмування, або переоцінюють свої знання і, будучи студентами, вважають, що все знають.

Вартим уваги є дослідження Л.В. Гришко (L.V. Gryshko), яка на основі проведеного ґрунтовного аналізу рівня знань студентів-першокурсників вказує, що в системі вищої освіти при підготовці інженерів-програмістів з метою розв’язання проблеми адаптації студентів до умов навчання у вищій школі і підвищення рівня знань зі шкільного курсу інформатики, пропонується до процесу навчання основ програмування включити пропедевтичний (вступний) курс, у якому передбачено навчання основ алгоритмізації і програмування з використанням мови програмування високого рівня Pascal, а з міркувань перспективності застосування програмування у професійній діяльності в курсі з основ програмування треба навчати студентів програмувати мовою C [6].

Якщо ж в Інтернеті пошукати відповідь на запитання, яку мову краще вивчати початківцеві, то більшість порад від “експертів” звучать приблизно так: “Якщо хочете працювати безпосередньо з залізом, писати драйвери й

додатки з максимальною продуктивністю, то підійдуть тільки С чи С++; якщо додатки для мобільних пристроїв, то Java, Objective C, С#; для веб-серверів — Go, Python і PHP; для веб-додатків — JavaScript” тощо. Або: “Якщо опиратися на свій досвід, то найкомфортнішою є мова С#, і з неї я б почав”, або ще: “Треба врахувати такі фактори, як ринок праці, довготривалі перспективи...” [7]. Можливість використання в навчанні мови Pascal вони категорично відхиляють, аргументуючи це передусім тим, що мова вже мертва (певною мірою це впливає й на ставлення студентів до неї). Але ці “експерти” мабуть забули, що професіоналами вони стали не одразу, а починали опановувати програмування на основі простих спеціальних навчальних мов, наприклад, Basic чи Pascal. І тільки зрідка можна натрапити на конструктивні думки [8]: “Початківцям в першу чергу треба не вчити саму мову, а вчитись правильно складати алгоритми, а вже потім незалежно, яку мову той чи інший обрав, — складати програми”, “Треба починати з легшого: швидкі результати в початківців мотивують їх боротися з проблемами далі”.

Мета: обґрунтувати переваги використання мови Pascal і недоцільності використання мови С як першої в навчанні інженерів-програмістів.

Завдання:

- визначити рівень підготовки студентів-першокурсників з програмування;
- на основі офіційних статистичних даних виявити на фоні популярності інших мов тенденцію використання мови Pascal у світі й Україні;
- проаналізувати можливості спеціальних навчальних мов програмування і мови системного програмування С як першої навчальної;
- вказати труднощі, які виникли на початкових етапах вивчення алгоритмізації і програмування на базі мови С.

Результати досліджень. За даними внутрішнього анкетування, проведеного на початку 2015-16 і 2016-17 навчальних років, 32,5% (27 із 83 опитаних студентів-першокурсників) і 30,6% (30 із 98 опитаних) у школі чи самостійно зовсім не вивчали програмування, 10,8% (9 студентів) і 29,6 % (29 студентів) лише ознайомилися (це суб'єктивна оцінка студентів) з однією чи кількома мовами програмування, 0% і 2,04% (2 студенти) відповідно вільно програмують. У попередні роки проводилося звичайне опитування, яке показувало, що мови програмування в школі вивчали від третини до половини студентів у кожній групі. Таким чином для багатьох студентів вивчення програмування починається тільки у виші. Як показав багаторічний досвід, багато тих, хто вивчав у школі програмування, мають поверхові, не систематизовані знання, програмують хаотично, хоч вважають, що вони програмують добре (багатьох з них навчити алгоритмізації і навчити, а по суті перевчити, програмувати за принципами структурного підходу набагато важче, ніж тих, хто починає вивчення предмету з нуля).

За даними досліджень рейтингів мов у світі відповідно до ТЮВЕ-індексів (визначають на основі пошукових запитів щодо назви мови за припущення, що між кількістю знайдених веб-сторінок і кількістю користувачів існує кореляційний зв'язок) [10], рейтинг Delphi/Object Pascal від лютого 2014, коли він був найнижчим, по 2017 (дані взято за лютий кожного року) поступово зростає (0,522% від загальної кількості запитів (20 місце), 1,574 (11), 2,163 (10), 2,479 (9); у березні 2017 — 2,544 (9); це свідчить про те, що мова не зовсім мертва), а рейтинг мови С спадає, залишаючись все одно високим (18,334% (1 місце), 16,488 (1), 15,594 (2), 8,445 (2); у березні 2017 — 7,742 (2)). Відповідно до ТЮВЕ-індексів протягом останніх 15 років найпопулярнішими мовами програмування є Java і С.

Опитування українських програмістів (2014-2017) [11] щодо використання мов програмування дали дещо інші результати: частка розробок мовою Pascal/Delphi зменшується (1,35% (11), \approx 1% (12), <1% (13), <1% (16)) і мовою С теж зменшується (1,65% (9), 1,4% (9), 1,3%(9), 1,1% (12)). В Україні програмісти найчастіше (з 2010) використовують мови Java, C#, PHP.

Як засіб навчання алгоритмізації і як першу мову програмування для початківців (з орієнтацією на непрограмістів) спочатку було розроблено багатоцільову мову символічних інструкцій Basic (1963). Щоб задовольнити вимоги структурного програмування, у подальшому було розроблено версії Turbo Basic, Quick Basic, Power Basic. Сучасними похідними мовами є Visual Basic, VBA, Visual Basic.NET. Але для тих, хто звик програмувати мовою Basic, перехід до інших мов викликає певні труднощі (зокрема простота динамічного виділення пам'яті під масиви в Basic).

Відповідно до принципів структурного програмування і з метою вироблення у студентів-програмістів хорошого стилю програмування Ніклаус Вірт (програміст і теоретик програмування) розробив навчальну мову Pascal (1970). У цій мові впроваджено жорстку перевірку типів, що дає змогу виявляти багато помилок на етапі компіляції. Програміст легко може описувати свої типи даних. Синтаксис мови Pascal простий і легко запам'ятовується, має достатньо засобів для розв'язування класичних завдань і опанування алгоритмізації. Програмний код легко читається, його можна передавати від одного програміста до іншого для вдосконалення й супроводу; у зв'язку з цим у багатьох випадках у літературі мову Pascal чи наближену до неї використовують як метамову для подання алгоритмів. Мова Pascal не потребує багато часу на вивчення, що дає можливість змістити акценти з її

синтаксичних і семантичних особливостей на вивчення методів побудови й аналізу алгоритмів, зосередитися на реалізації своїх структур даних. Крім того, Pascal є базою для вивчення ряду інших мов, оскільки отримані навички дають змогу легко перейти на будь-яку сучасну мову програмування. У роботі [5] відслідковується перехід на основі навичок, одержаних при вивченні мови Pascal, до об'єктно-орієнтованого програмування: “Найбільш природним і строгим перехід від процедурного програмування до ООП вбачається з використанням мови Pascal, з використанням ланцюжка понять: типи — процедури — передавання даних — запис (структура як властивість) — об'єкт (структура як властивість + процедура як метод)”.

На основі мови Pascal було розроблено об'єктно-орієнтований Object Pascal (Delphi), останні версії якого розраховані на розробку кросплатформних додатків для Windows 10, Mac OS, iOS, Android (версію Delphi 10.1 випущено в 2016 [9]). Протягом багатьох років Pascal широко використовувався не тільки для навчання, а й при розв'язуванні різноманітних прикладних задач. У наш час він значно втратив свою популярність як мова програмування і зовсім необгрунтовано втрачає популярність як мова навчання у вишах.

Мова програмування C (1972) розроблена не як навчальна мова і навіть не як мова загального призначення, а як мова системного професійного програмування. Проте нею можна написати практично будь-яку прикладну програму, і вона використовується для розв'язування найрізноманітніших задач. Продовженнями мови C є такі популярні у наш час мови, як C++, Objective-C, C#, Java. Мова C — це мова програмування середнього рівня, вона об'єднує кращі риси мов високого рівня з можливостями Асемблера. Як мова середнього рівня C дає

можливість маніпулювати бітами, байтами й адресами, тобто тими базовими елементами даних, з якими працює комп'ютер. Проте на відміну від більшості мов високого рівня, в С майже повністю відсутній контроль помилок при виконанні програми. Наприклад, не перевіряється порушення границь масивів. Відповідальність за такі помилки повністю покладається на програміста. Синтаксис мови С досить складний. Також С не вимагає суворої сумісності формальних параметрів і аргументів функцій на відміну від мов програмування високого рівня, де, як правило, треба, щоб тип аргумента відповідав типу параметра. У мові С аргумент може мати майже будь-який тип, якщо його можна розумно перетворити на тип параметра. Компілятор С автоматично виконує всі види перетворень.

При використанні мови програмування С (на відміну від Pascal) як базової при вивченні курсу “Алгоритмізація та програмування” на початкових етапах виникло ряд труднощів, пов’язаних з її особливостями (середовище Microsoft Visual Studio): студенти повинні пам’ятати багато нюансів, відволікаючись при цьому від головного — алгоритмізації. Розглянемо найосновніші з них.

1. Оскільки всі функції при програмуванні мовою С є бібліотечними, то треба слідкувати за тим, щоб відповідні бібліотеки були підімкнуті командою препроцесора `#include`. Крім того, для використання числових констант (числа π , e і пов’язані з ними величини) бібліотеки `math` треба препроцесорові вказати директиву `#define _USE_MATH_DEFINES` ще до директиви підмикання бібліотеки `#include <math.h>`. У мові Pascal більшість функцій стандартні і не потребують підмикання бібліотек; числу π відповідає константа `pi`, e — `exp(1)`.

2. У програмах мовою С дуже важливим є правильне використання форматів вводу й виводу при зверненні до

функцій `scanf` і `printf`. Від цього залежить і правильність обчислень, у яких беруть участь вхідні дані, і правильність візуалізації результатів — програма може порахувати все правильно, а виведення за невідповідним до типу одержаного значення форматом відобразить зовсім інший результат; наприклад, `printf("%d", 5.)` і `printf("%f", 5)` дадуть результат 0, `printf("%c", 5.)` — пробіл, `printf("%c", 5)` — символ `␣`, `printf("%s", 5.)` — нульовий вказівник (`null`), а `printf("%s", 5)` — аварійно закінчить виконання програми. Крім того, оскільки формати виведення вказуються на відстані (у першому параметрі — рядку формату) від аргумента (змінної чи виразу), то не завжди вистачає уважності швидко виявити помилку. У Pascal формати при введенні не використовуються, а при виведенні не обов'язкові, їх можна вказати для красивого оформлення результатів, і формат міститься зразу після виразу.

3. Математичні функції бібліотеки `math`, залежно від типів, мають різні імена (стандарт C99). Так, для типу `double` використовуються “математичні” імена функцій, а для типів `float` і `long double` імена містять постфікси `f` і `l` (наприклад, `double sin(double x)`, `float sinf(float x)`, `long double sinl(long double x)`). Також часта помилка у студентів — сплутують використання функцій `fabs(x)` (результатом є дійсне число) і `abs(x)` (результатом є ціле число; дробова частина відкидається). На відміну від C у мові Pascal деякі функції можуть працювати з даними різних типів, наприклад, функція `abs` для цілого аргумента дає цілий результат, а для дійсного — дійсний; студенти тільки часом сплутують функції `sqrt` — квадратний корінь і `sqg` — піднесення до квадрата.

4. У програмі мовою C обчислюється будь-який вираз (включаючи ділення на 0, знаходження кореня з від'ємного числа тощо — результат відповідно буде `INF` і `NAN`; переповнення також дає `INF`). При програмуванні

математичних виразів треба перед обчисленнями перевіряти, чи значення аргумента є допустимим. У частині студентських програм було порушено порядок дій: спочатку обчислювалося значення виразу, а потім перевірялася коректність обчислень і в разі необхідності видавалося повідомлення про недопустиме значення. При цьому основним твердженням було: “Але ж програма працює”. Щоб довести студентів, що програма некоректна, хоча щось і виконує, треба докласти багато зусиль. У даній ситуації Pascal-програма видала б повідомлення про ділення на нуль, від’ємне значення під коренем, недодатній підлогарифмічний вираз, про переповнення і аварійно закінчила б роботу.

5. Для подання складних умов, наприклад, $x \in [-5; 8)$ треба скористатися логічним виразом $(x >= -5 \ \&\& \ x < 8)$. Якщо умову записати з використанням подвійної $-5 < = x < 8$ нерівності, то компілятор не вкаже на помилку — цей вираз буде обчислюватися як $(-5 < = x) < 8$: спочатку обчислиться вираз $(-5 < = x)$, і результатом залежно від значення x буде 0 чи 1, а потім $0 < 8$ чи $1 < 8$, кожен з яких дасть значення 1 (“істина”). І знову аргумент: “Але ж програма працює”. У Pascal це була б синтаксична помилка.

6. При використанні операції ділення в мові C треба пам’ятати, що ділення цілих величин дає цілий результат (відкидається дробова частина, навіть якщо вона близька до 1). Тому, щоб зрозуміти, що при обчисленні значення виразу результат не відповідає очікуваному, треба паралельно провести розрахунки, наприклад, в електронних таблицях Excel. У мові Pascal для цілочисельного і дійсного ділення використовуються різні операції (відповідно `div` і `/`).

7. Змішування типів даних в одному виразі компілятор мови C, на відміну від компілятора Pascal, не вважає

помилкою. Наприклад, у виразах можна вільно змішувати змінні символічного і цілого типів. При обчисленні виразів, у яких змішано дані різних типів, компілятор автоматично виконує неявне перетворення типів за спеціальними правилами. Це може бути як зручним, так і небезпечним, якщо змішування типів є випадковим. Тому при використанні мови C як базової обов'язково спочатку треба розібратися з перетворенням типів даних, оскільки нерозуміння цього може призвести до непередбачуваних результатів виконання коду. У ряді випадків неможливо обійтися без явного перетворення типів. Наприклад, при виконанні фрагменту `double f; int b=32767; f=100*(float)b*b;` без перетворення змінної `b` до типу `float` буде втрачено результат, оскільки значення $100*b*b$ більше від максимально можливого значення даних типу `int`.

8. Порівняно з іншими мовами у мові C є досить велика кількість операцій (більше 35). Різні операції позначаються тими самими символами або їхнім подвоєнням. Наприклад, `*` позначає множення, вказівник, розіменування вказівника. Часто при програмуванні логічних виразів замість операції порівняння `==` студенти записують операцію присвоювання `=`, і таку помилку досить важко виявити.

9. Після розгляду на лекціях нестандартних можливостей використання оператора `for`, пов'язаних з тим, що `for`-цикли не обмежуються тільки випадком арифметичної прогресії, студенти старалися запрограмувати весь вміст тіла оператора в полі умови чи полі збільшення параметра, не зважаючи на те, що при поданні матеріалу наголошувалося, що включати в заголовок циклу обчислення, які не мають відношення до ініціалізації й збільшення чи зменшення значення параметра циклу, вважається поганим стилем, знижує

читабельність програмного коду, призводить до збільшення помилок.

Таких на перший погляд незначних труднощів при програмуванні мовою С виникає ще досить багато. Вони відволікають від алгоритмізації. Щоб програма, написана мовою С, правильно і надійно працювала, потрібно глибоко занурюватися в деталі мовних засобів. Звичайно, можна в першому семестрі при реалізації навчальних алгоритмів скористатися якоюсь досить обмеженою підмножиною мови С. Наприклад, розглядати тільки типи `int` і `double` і відповідно тоді буде менше проблем з форматами введення й виведення і перетворенням типів, не розглядати нестандартні можливості циклу `for` тощо. Але в подальшому все рівно доведеться повернутися до глибшого вивчення базових понять мови С.

Для початківців мова Pascal — це ніби каліграфія, а мова С — стенографія з багатьма умовними позначеннями. Головне, щоб студент не зневірився у своїх силах і не втратив бажання вчитися взагалі.

Першою навчальною мовою при вивченні алгоритмізації і програмування обов'язково має бути мова з простим синтаксисом, щоб студенти могли зосередитися на головному — алгоритмізації. На основі проведених досліджень можна зробити такі ґрунтовні висновки:

— проведені анкетування й опитування показали, що досить великий відсоток студентів (майбутніх інженерів-програмістів) у школі не вивчали програмування;

— на основі статистичних даних (ГІОВЕ-індексів) у світі виявлено зростання протягом останніх трьох років рейтингу об'єктного розширення мови Pascal — Delphi/Object Pascal, хоч в Україні за даними багаторічних досліджень популярність Pascal/Delphi серед програмістів дуже низька і спадає;

— аналіз можливостей спеціальних навчальних мов програмування і мови системного програмування С як першої навчальної показав, що використання спеціальної, вже класичної, навчальної мови Pascal (нову навчальну мову, яка б повніше відповідала сучасним вимогам, ще не розроблено) не потребує багато часу на її засвоєння, сприяє зосередженню на алгоритмізації, від неї легко переходити до вивчення інших сучасних рейтинговіших мов програмування;

— використання мови програмування С як базової на початкових етапах викликає ряд труднощів, пов'язаних з особливостями її синтаксису і відсутністю контролю помилок при виконанні програми, відволікає від алгоритмізації, і тому не є доцільним.

ЛІТЕРАТУРА

1. Ящик О.Б. Методика навчання алгоритмізації та програмування старшокласників на рівні поглибленого вивчення інформатики: дис. на здобуття наук. ступеня канд. педагогічних наук: спец. 13.00.02 “Теорія та методика навчання (інформатика)” [Текст] / Олександр Богданович Ящик; НПУ ім. М.П. Драгоманова. — Київ, 2016. — 352 с.
2. Пилипчук О.П. Про вибір мови програмування [Електронний ресурс] / О.П. Пилипчук // Комп'ютер у школі та сім'ї. — 2014. — № 2. — С. 22 - 23. — Режим доступу: http://nbuv.gov.ua/UJRN/komp_2014_2_6
3. Яку мову програмування вивчати у школі (матеріали для дискусії) [Електронний ресурс] // Комп'ютер у школі та сім'ї. — 2013. — № 7. — С. 14 - 18. — Режим доступу: http://nbuv.gov.ua/UJRN/komp_2013_7_5
4. Яку мову програмування вивчати у школі (матеріали для дискусії) [Електронний ресурс] // Комп'ютер у школі та

- сім'ї. — 2013. — № 8. — С. 9-18. — Режим доступу: http://nbuv.gov.ua/UJRN/komp_2013_8_3
5. Лапінський В.В. Проблема вибору першої мови програмування — сьогоднішнє бачення [Електронний ресурс] / В.В. Лапінський // Комп'ютер у школі та сім'ї. — 2014. — № 1. — С. 14-17. — Режим доступу: http://nbuv.gov.ua/UJRN/komp_2014_1_4
6. Гришко Л.В. Методична система навчання основ програмування майбутніх інженерів-програмістів: автореф. дис. на здобуття наук. ступеня канд. педагогічних наук: спец. 13.00.02 “Теорія та методика навчання (інформатика)” [Текст] / Людмила Веніамінівна Гришко; НПУ ім. М.П. Драгоманова. — Київ, 2009. — 34 с.
7. Какой язык программирования лучше выбрать первым для изучения новичку? (29.03.2016) [Електронний ресурс]. — Режим доступу: <https://tproger.ru/experts/16/>
8. Подоба В. Програміст Початківець: Яку мову, технологію, напрямок обрати? (01.02.2014) [Електронний ресурс] / В. Подоба. — Режим доступу: <http://www.vitaliyподоба.com/2014/02/newbie-programmer-what-language-to-choose/>
9. Delphi Overview [Електронний ресурс]. — Режим доступу: www.embarcadero.com/products/delphi
10. ТІОБЕ Index [Електронний ресурс]. — Режим доступу: <https://tiobe.com/tiobe-index/>
11. Опрос по языкам #8 (23.01.2017) [Електронний ресурс]. — Режим доступу: <https://dou.ua/lenta/articles/language-rating-jan-2017/>

Транслітераційний переклад використаних джерел

1. Yashchuk A. V. Metodyka navchannia alhorytmizatsii ta prohramuvannia starshoklasnykiv na rivni pohlyblenoho vyvchennia informatyky [Methods of learning algorithms and programming at of senior pupils in-depth study of computer

- science] (Candidate dissertation). (2016). National Pedagogical Dragomanov University, Kyiv (in Ukr.)
2. Pilipchuk, A.P. (2014, March). Pro vybir movy prohramuvannia [On the choice of programming language]. *Komp'iuter u shkoli ta sim'i (Computer in school and family)*, 2(114), 22 - 23. Retrieved from http://nbuv.gov.ua/UJRN/komp_2014_2_6 (in Ukr.)
3. Yaku movu prohramuvannia vuvchaty u shkoli (materialy dlia dyskusii) [What programming language to learn in school (materials for discussion)] (2013, October). *Komp'iuter u shkoli ta sim'i Computer in school and family*, 7(111), 14-18. Retrieved from http://nbuv.gov.ua/UJRN/komp_2013_7_5 (in Ukr.)
4. Yaku movu prohramuvannia vuvchaty u shkoli (materialy dlia dyskusii) [What programming language to learn in school (materials for discussion)] (2013, December). *Komp'iuter u shkoli ta sim'i (Computer in school and family)*, 8(112), 9-18. Retrieved from http://nbuv.gov.ua/UJRN/komp_2013_8_3 (in Ukr.)
5. Lapinsky, V.V. (2014, February) Problema vyboru pershoi movy prohramuvannia — sohodnishnie bachennia [Problem of choosing the first programming language — contemporary vision]. *Komp'iuter u shkoli ta sim'i (Computer in school and family)*, 1(113), 14-17. Retrieved from http://nbuv.gov.ua/UJRN/komp_2014_1_4 (in Ukr.)
6. Gryshko, L.V. Metodychna systema navchannia osnov prohramuvannia maibutnikh inzheneriv-prohramistiv [Methodical system of teaching the basics of programming for future programming engineers] (Candidate dissertation). (2009). National Pedagogical Dragomanov University, Kyiv (in Ukr.)
7. Kakoi iazyk programmirovaniia luchshe vybrat pervym dlia izucheniia novichku? [Which programming language is best to

- choose first for learning a beginner?]) (2016, March). Retrieved from <https://tproger.ru/experts/16/> (in Russ.)
8. Podoba V. (2014, February). Prohramist Pochatkivets: Yaku movu, tekhnolohiiu, napriamok obraty? [Newbie-programmer: what language, technology, direction to choose?] Retrieved from <http://www.vitaliypodoba.com/2014/02/newbie-programmer-what-language-to-choose/> (in Ukr.)
9. Delphi Overview(2017, March). Retrieved from www.embarcadero.com/products/delphi
10. TIOBE Index(2017, March). Retrieved from <https://tiobe.com/tiobe-index/>
11. Opros po iazykam #8 [Language Interview # 8] (2017, January). Retrieved from <https://dou.ua/lenta/articles/language-rating-jan-2017/> (in Russ.)

Larysa KUBLII

Candidate of Technical Sciences, Associate Professor at the Department of Automation of Power Processes and Systems Engineering
National Technical University of Ukraine “Ihor Sikorskii Kyiv Polytechnic Institute”

e-mail: kublii_l_i@ukr.net

THE BASIC LANGUAGE IN STUDYING ALGORITHMISATION AND PROGRAMMING BY STUDENTS

***ABSTRACT.** The article deals with the problem of choosing the basic programming language for the first-semester students' study of “Algorithmization and Programming” as a part of their training as software engineers. The use of Pascal and C basic languages has been analysed taking into account*

the capabilities of these languages and the level of school knowledge of the first-year students.

The C programming language peculiarities distracting students from mastering the basics of algorithmization and developing good programming style at the initial stages have also been pointed out.

Capacity analysis of special teaching programming languages and C system programming language has shown that special classic teaching Pascal language (a new teaching language which would better meet modern requirements has not yet designed yet) does not tube much time to master facilitates algorithmization and study of other modern programming languages with higher rating.

The use of C programming language as basic one at the initial stages creates a number of difficulties related to its specific syntax and absence of error control during the fultiment of a programme, prevents algorithmization and therefore is not advisable.

Key words: *training of software engineers, algorithmization, basic programming language, Pascal programming language, C programming language, structured programming.*

Лариса Ивановна КУБЛИЙ

кандидат технических наук, доцент кафедры
автоматизации проектирования энергетических процессов
и систем,
НТУУ “Киевский политехнический институт имени Игоря
Сикорского”

e-mail: kublii_l_i@ukr.net

БАЗОВЫЙ ЯЗЫК ПРИ ИЗУЧЕНИИ СТУДЕНТАМИ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Аннотация. В статье рассмотрена проблема выбора базового языка программирования при изучении студентами дисциплины "Алгоритмизация и программирование" в первом семестре первого года подготовки инженеров-программистов. Проанализировано использование языков Pascal и C как базовых с учетом возможностей этих языков и уровня школьной подготовки студентов-первокурсников.

Указано особенности языка программирования C, которые на начальных этапах отвлекают студентов от постижения основ алгоритмизации и выработки хорошего стиля программирования.

Анализ возможностей специальных учебных языков программирования и языка системного программирования C как первых учебных показал, что использование специального уже классического, учебного языка Pascal (новый учебный язык, который полнее отвечал современным требованиям, еще не разработан) не требует много времени на его усвоение, способствует сосредоточению на алгоритмизации, от него легко переходить к изучению других современных с большим рейтингом языков программирования. Использование языка программирования C как базового на начальных этапах вызывает ряд трудностей, связанных с особенностями его синтаксиса и отсутствием контроля ошибок при выполнении программы, отвлекает от алгоритмизации, и поэтому не является целесообразным.

Ключевые слова: подготовка инженеров-программистов, алгоритмизация, базовый язык, язык программирования Паскаль, язык программирования C, структурное программирование.