

## ДО ПИТАННЯ ВИВЧЕННЯ ТЕМИ "АЛГОРИТМИ ТА ВИКОНАВЦІ" У КУРСІ "ІНФОРМАТИКА" ПОЧАТКОВОЇ ШКОЛИ

*У статті автор, на основі аналізу чинних підручників з курсу "Інформатика" для початкової школи, розкриває методичні особливості введення основних понять змістової лінії "Алгоритми і виконавці", аналізує труднощі, що виникають в учнів під час вивчення даної теми.*

**Ключові слова:** алгоритм, способи подання алгоритмів, команда, середовище виконавця, труднощі при формуванні уявлення про алгоритм.

**Постановка проблеми та її зв'язок із важливими науковими чи практичними завданнями.** Впровадження пропедевтичного курсу "Інформатика" у початкову школу не могло не позначитись на змісті вищої педагогічної освіти на факультетах початкового навчання педагогічних університетів. З 2012-2013 н.р. за напрямом підготовки 0101 "Педагогічна освіта", спеціальності 6.010102 "Початкова освіта" введено нову навчальну дисципліну "Методика навчання інформатики". Одним із головних завдань якої є забезпечення ґрунтового розуміння студентами методичних ідей початкового курсу інформатики, використання методів і засобів нових інформаційних технологій. У зв'язку з цим набуває актуальності питання дослідження сучасних методичних підходів до реалізації змісту навчання курсу "Інформатика" у початковій школі.

**Аналіз останніх досліджень і публікацій.** Різні проблеми пропедевтики навчання інформатики у початковій школі в Україні знайшли своє вирішення у працях науковців та дослідників, зокрема:

– обґрунтовані авторські концепції методичних комплексів "Сходинки до інформатики", що реалізують вимоги Державного стандарту початкової загальної освіти (Н. Морзе, В. Вембер, Н. Саражинська, О. Коршунова, І. Зарецька, М. Корнієнко, С. Крамаровська, Г. Ломаковська, Г. Проценко, Й. Ривкінд, Ф. Рівкінд);

– розкриті методологічні основи застосування комп'ютера в початковій школі та проведений аналіз чинних підручників "Сходинки до інформатики" на відповідність вимогам до сучасного підручника даної вікової групи (В. Шакоцько, О. Кивлюк);

– обґрунтовані деякі шляхи удосконалення змісту, методів, засобів та організаційних форм навчання інформатики шляхом впровадження пропедевтики змістово-методичної лінії формалізації та моделювання на основі включення спеціальних завдань (М. Глазун, Н. Морзе); методики використання опорних схем, організації розробки міні проектів (О. Коршунова); формувального оцінювання (Н. Морзе, В. Вембер, О. Барна); завдань на розвиток мислення учнів (В. Вембер) та ін.

Проте питання узагальнення методичних підходів опрацювання змістових ліній пропедевтичного курсу інформатики у початковій школі не стало предметом окремого дослідження. Вирішення вказаного завдання сприятиме наповненню лекційного курсу навчальної дисципліни "Методика навчання інформатики" на факультетах початкового навчання педагогічних вишів, формуванню у студентів варіативного бачення щодо введення понять, способів дій, використання прийомів та методів навчання, методів та засобів інформаційних технологій.

**Мета дослідження** – на основі аналізу чинних підручників з курсу "Інформатика" початкової школи розкрити методичні особливості введення основних понять змістової лінії "Алгоритми і виконавці", проаналізувати труднощі, що виникають в учнів під час вивчення даної теми.

**Результати дослідження.** Ознайомлення учнів з алгоритмами та їх виконавцями передбачає розуміння на інтуїтивному рівні понять: виконавця, його середовища, команди, системи команд виконавця, алгоритму, отримання перших уявлень про основні алгоритмічні структури: слідування, розгалуження та повторення, формування навичок виконувати готові алгоритми, а також складати прості алгоритми для виконавців, які працюють у певному зрозумілому для відповідної вікової категорії комп'ютерному середовищі, використовуючи просту систему їхніх команд. Діти отримують уявлення про способи подання алгоритмів: словесний; алгоритмічну мову (як систему команд виконавця комп'ютерного середовища) та графічний (побудова алгоритмів на основі блок-схем).

У діючих підручниках "Сходинки до інформатики" для 2-3 класів запроваджено різні підходи щодо введення понять та ознайомленням із способом подання алгоритмів. Розглянемо деякі з них (див рис.1,2). І підхід (за О. Коршуновою) характеризується послідовним ознайомленням учнів із словесним поданням

алгоритму (2-3 класи) та мовою виконавця комп'ютерного середовища **Scratch** (3 клас). Уявлення про ці способи виробляються на прикладах інтуїтивно.

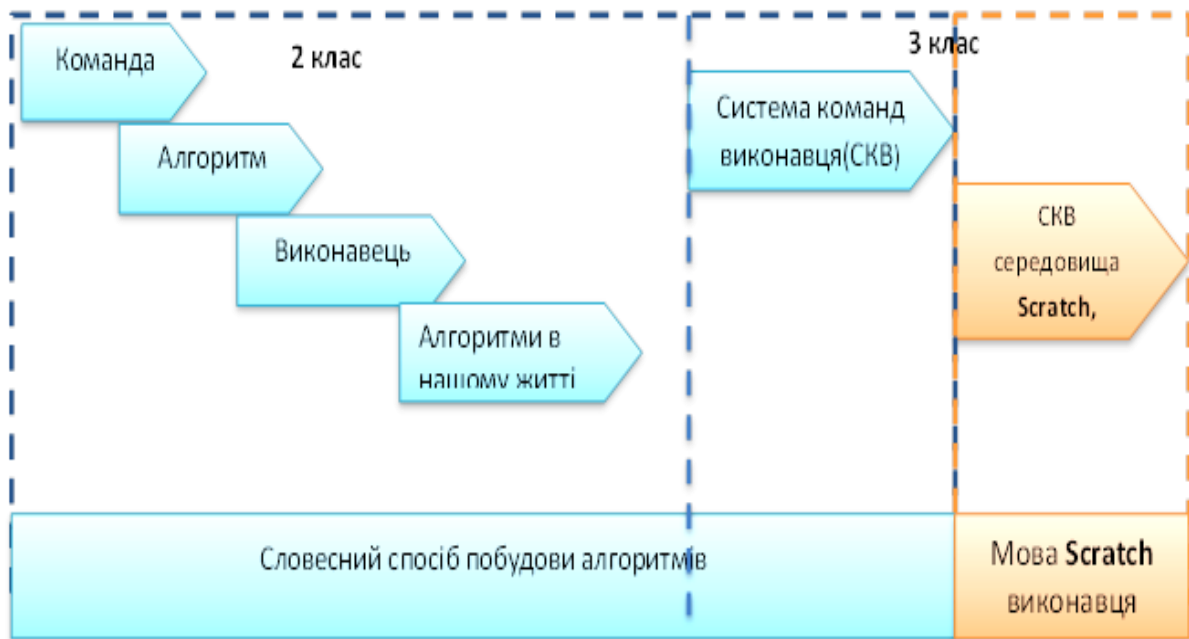


Рис. 1. Вивчення теми «Алгоритми та виконавці» за О. Коршуною

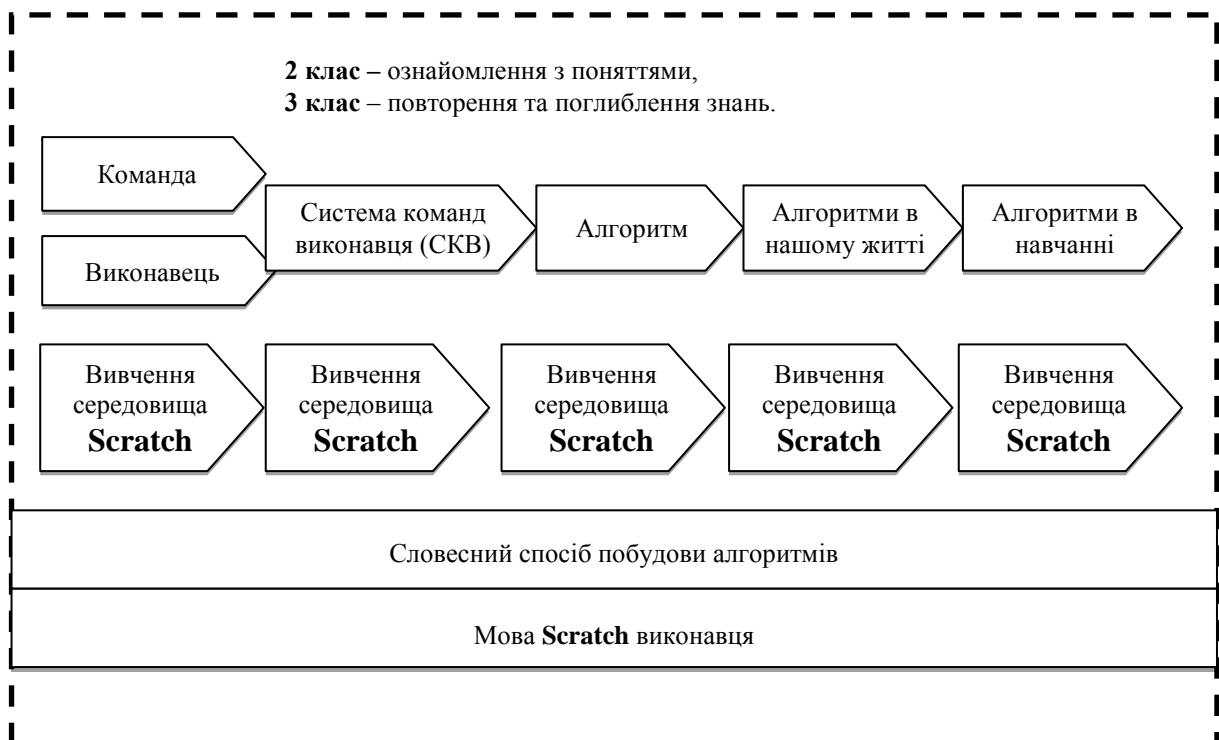


Рис. 2. Вивчення теми «Алгоритми та виконавці» за Г. Ломаковською, Г. Проценко, Й. Ривкіндом, Ф. Ривкінд.

У II підході (за Г. Ломаковською та ін.) способи побудови алгоритмів словесний та мовою виконавця **Scratch** вивчаються паралельно. Авторами реалізовано принцип концентричності, що передбачає повторення та вивчення уже знайомих понять у 3 класі на більш високому рівні (повторення, розширення та поглиблення знань). Так, у 3 класі описово дається уявлення про словесний спосіб подання команд, його властивості й інші способи (жести, сигнали); розкривається новий зміст понять команда, алгоритм.

Уявлення про алгоритм та поняття з ним пов'язані формуються невіддільно від способів подання алгоритмів. Так, на основі **словесного способу подання алгоритму** діти починають ознайомлення з

"алгоритмом" та правилами його створення. Спочатку дається уявлення про "команду": "Спонукальне речення спонукає до негайної дії. Воно є наказом, командою" [6].

"Команда – це речення, що спонукає до дії. Той, хто виконує команди, називається **виконавцем**" [7].

Пізніше (3 клас) за методичним підходом Г. Ломаковської та ін. уявлення про команду розширюється: "Команда – наказ, вказівка виконати певні дії" [9]. Звертається увага, що подання команд у формі спонукальних речень – це лише один із способів подання команд, який називається словесним. На основі конкретно-індуктивного методу дітей ознайомлюють з іншими способами подання команд: світловими сигналами (світлофором); звуковими сигналами (дзвінком у школі); жестами (міліціонером-регулювальником); подвійним клацанням лівою клавішею миші на виділеному об'єкті (при запуску програм на виконання). За методикою О. Коршунової [13], крім словесного способу подання команд неявно дається уявлення й про інші способи: мовою стрілок; малюнками; за допомогою скороченого запису команд. За допомогою названих умовних позначень команд учні складають й відповідні алгоритми.

У чинних підручниках поняття "**виконавця**" розглядається у залежності від порядку його введення, тобто спирається на поняття введені раніше. Так, виконавець – це об'єкт (людина, тварина, технічний пристрій), який виконує команди (за Г. Ломаковською та ін.), або об'єкт (людина, технічний пристрій), який виконує алгоритм (за О. Коршуновою).

#### **Вимоги до команд виконавця**

Всі команди повинні бути:

- 1) точними;
- 2) зрозумілими;
- 3) виконуваними.

Ці властивості характеризують й поняття "система команд виконавця".

Властивості "зрозумілості" та "виконуваності" діти усвідомлюють інтуїтивно. Наприклад, пояснюють, що виконавець водій розуміє (знає) й може виконувати лише команди, передбачені правилами дорожнього руху. У казці "Івасик Телесик" головний герой припливав до берега (виконував команди), коли впізнавав (розумів) голос матері: "Івасику Телесику приплинь, приплинь до бережка!" тощо. Для технічних пристроїв властивість "зрозумілості" є умовною (пристрої не можуть розуміти) і означає розпізнання тих команд, які належать до множини виконуваних ним команд (системи команд виконавця).

Властивість "точності" команд можна пояснити на основі аналогії з інструкцією, правилом, рецептом які людині даються для виконання різних життєвих завдань (наприклад, рецепт приготування чаю, інструкція експлуатації технічного пристрою): чим точніше описані правила, тим швидше людина опанує їх і буде ефективноше застосовувати.

Поняття "алгоритм" є не означуваним, основним поняттям інформатики. Термін "алгоритм" походить від імені давнього філософа і математика із Хорезму – Аль-Хорезмі (IX століття). Він описав правила арифметичних дій над багатоцифровими числами. У сучасних підручниках поняття алгоритму вводиться описово, при цьому нестрогі означення вказують на ті чи інші необхідні ознаки алгоритму.

**Алгоритм** – це послідовність точних, зрозумілих *команд* виконавцю, для вирішення якогось завдання [7].

Усі дії записані у вигляді команд і у певній послідовності, а їх виконання призводить до розв'язання певної задачі. Такий план дій називають **алгоритмом** [8].

**Алгоритм** – це послідовність команд. В алгоритм можуть входити лише ті команди, які виконавець може виконати, тобто команди із системи команд цього виконавця [9].

**Алгоритмом** називають порядок команд, що дає змогу виконати певну задачу, а виконавцями – тих хто їх виконує [6, 5].

Таким чином, крім уже відомих властивостей команд (зрозумілість, виконуваність, точність) додаються нові властивості-ознаки – послідовність команд (команди повинні виконуватись у порядку їх слідування у записі алгоритму, зміна порядку слідування або пропуск команди призведе до неправильного результату); результативність (виконавши алгоритм, ми матимемо певний результат); правильність, що означає досягнення мети (результат повинен бути "розв'язком" задачі). Розумінню названих властивостей сприяє виконання завдань на складання алгоритмів, їх виконання, розміщення даних команд у необхідній послідовності, що призведе до вирішення задачі (2 клас) [7,8]; виправлення помилок у послідовності команд, виявлення зайвої або пропущеної команди (за методичним підходом Г. Ломаковської, Г. Проценко, Й. Ривкінда, Ф. Ривкінд такі завдання пропонуються у 3 класі [9]).

*Правила подання алгоритму словесним способом* можна сформулювати наступним чином: алгоритм має назву, у якій розкривається його призначення, наприклад "Алгоритм розв'язування рівнянь", "Алгоритм визначення будови слова", "Алгоритм "Банан"" тощо. Усі команди нумеруються, що визначає послідовність їх виконання.

З навчальною метою передбачається невелика кількість команд у завданнях та прикладах. Наприклад.

#### **Алгоритм відкриття дверей ключем:**

- 1) дістати ключ;
- 2) вставити ключ в отвір дверей;
- 3) повернути ключ за годинниковою стрілкою;
- 4) вийняти ключ.

**Система команд виконавця** визначає множину команд (елементарних дій), які виконавець розуміє і здатний їх виконувати. Уявлення про систему команд виконавця за підручником [7] формується конкретно-індуктивним способом. Приклади виконавців та їх систем команд ілюструються за допомогою таблиці. Таке унаочнення сприятиме усвідомленню того, що кожен виконавець може виконати лише команди із власної системи команд, а команди, що не входять до його системи, будуть не зрозумілі.


Інший підхід щодо усвідомлення поняття системи команд здійснений у підручнику [6]. Тут використаний метод порівняльного аналізу прикладів із застосуванням прийому абстрагування:

– Виконавці абстраговані від дійсності (введення позначень: Виконавець №1, Виконавець №2)

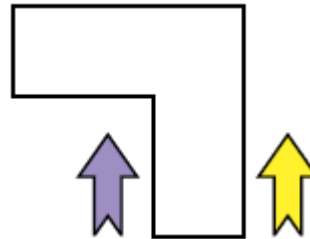
– Образи, на які спираються виконавці знайомі учням і мають незначну зовнішню відмінність.

– Система команд для кожного виконавця може добиратись самостійно і не співпадати із системою команд реального об'єкта, образ якого запозичений.

Це дає змогу підготувати ґрунт для розуміння комп'ютерного виконавця й системи його команд. Розглянемо приклад із підручника [4].

<p>Виконавець № 1</p> 	<p>Виконавець № 2</p> 
<p>Вміє виконувати команди:</p> <ul style="list-style-type: none"> <li>– Поверни праворуч.</li> <li>– Їдь уперед.</li> </ul>	<p>Вміє виконувати команди:</p> <ul style="list-style-type: none"> <li>– Поверни праворуч.</li> <li>– Поверни ліворуч.</li> <li>– Їдь уперед.</li> </ul>

Задача для виконавців – проїхати по маршруту.



**Алгоритм для виконавця №1**

1. Їдь уперед.
2. Поверни праворуч.
3. Поверни праворуч.
4. Поверни праворуч.
5. Їдь уперед.

**Алгоритм для виконавця №2**

1. Їдь уперед.
2. Поверни ліворуч.
3. Їдь уперед.

Таким чином, дія "поверни ліворуч" для другого виконавця є "елементарною" – для неї існує відповідна команда. Для першого виконавця для цієї дії не існує відповідної команди у системі команд, тому вона виконується трьома командами. На нашу думку, доцільними будуть запитання для учнів: Які команди містить алгоритм виконавця №1? Які команди містить алгоритм виконавця №2? Чому перший алгоритм містить більшу кількість команд? Учні мають усвідомити, що алгоритми для виконавців будуються лише з системи його команд виконавця. Виконавець не зможе виконати алгоритм написаний для іншого виконавця, бо не буде розуміти команд, які не належать до його системи команд. Діти повинні дійти висновку, що алгоритм будуватиметься лише з команд, які входять до системи команд виконавця.

У сучасній методиці навчання інформатики виділяють труднощі, пов'язані з формуванням чіткого уявлення про алгоритм:

– В уяві дитини є цілісний образ розв'язання завдання, а вимагається дати чіткий опис цього процесу у формі послідовності спонукальних речень.

– У задачах такого типу важко виділити "елементарні" дії, з яких складатиметься алгоритм.

– При виконанні алгоритмів, виконавець не повинен демонструвати елементи творчості – лише чітко виконувати команди у вказаному порядку [10, с. 18].

Для їх подолання методисти пропонують використання *рольового методу навчання*. Для цього виділяються ролі *пояснюючого* та *виконавця* алгоритму. Пояснюючий складає алгоритм, зачитує команди, виконавець програве накази. При неможливості виконання якоїсь команди, вона уточнюється чи деталізується до тих пір, поки не стане зрозумілою і виконуваною – "елементарною". Щоб виробити навички формального виконання алгоритму (виконавець не повинен розуміти суть проблеми чи завдання, намагатися додати власні дії, яких не має у записі алгоритму), вчителі-практики використовують прийом складання алгоритмів для виконавця-робота (програвання ролі робота, згодом перенесення його основних властивостей і на саме поняття виконавця).

Для цього потрібно актуалізувати знання учнів про Робота: "Ти знаєш що таке РОБОТ? Спробуй відповісти на такі питання: Він живий? (Ні) Він з дерева? (Ні) Він з металу?(Так) Він машина?(Так) У нього є мозок? (Ні) Він сам може думати? (Ні) Хто ж ним керує? (Людина) А як же людина керує роботом? (За допомогою комп'ютера). Отже, **робот** – це машина, зроблена з металу. Робот допомагає людям виконувати складну роботу. Людина керує роботом. Робот не може думати. Щоб керувати роботом, йому потрібно давати чіткі і зрозумілі **команди**".

Розглянемо гру "Виконай дії" [4]. Метою гри є створення колективного алгоритму для Робота з виконання деякого завдання. Правила гри можуть бути такими:

1. Команди алгоритму повинні бути чіткими і зрозумілими.
2. Кожну команду для алгоритму робот повинен програти.
3. Команда може уточнюватись і програтись доти, доки не стане виконуваною.
4. Здійснену послідовність команд роботом потрібно записати.

На роль робота-виконавця спочатку може обиратися вчитель. Наприклад. Учитель повідомляє дітям, що він на деякий час стане роботом і пропонує скласти алгоритм виконання завдання "Поливу квітів", які містяться на підвіконні. Там же стоїть склянка. А вода набирається з крану рукомийника. Діти повинні відповісти на запитання: "Якою буде перша команда?" Імовірно, що команди, запропоновані учнями будуть: "Набери у склянку води. Поливай квіти". Учитель повинен з гумором реагувати на невірні команди. Так, до першої команди можна продемонструвати, що робот не може дотягнутися ні до склянки, ні до води. Уточнення потребує команда "взьми склянку" – робот може показати дію, що хоче взяти склянку, яку йому хтось подасть. Тому потрібно уточнити звідки він візьме склянку. Якщо пропущені команда "Відкрий кран" – робот може тримати склянку під краном так і не набравши води. І т.д. Останньою повинна бути команда "Постав склянку на місце", бо без неї робот буде продовжувати тримати склянку. У результаті на дошці має бути записаний алгоритм.

#### **Алгоритм поливу квітів**

1. Підійди до підвіконня.
2. Візьми склянку з підвіконня.
3. Підійди до рукомийника.
4. Відкрий кран.
5. Набери з під крану у склянку воду.
6. Закрути кран.
7. Підійди до підвіконня.
8. Поливай квіти.
9. Постав склянку на її місце.

Важливе місце у системі тем "Алгоритми і виконавці" у чинних підручниках відведено виробленню уявлень про застосування алгоритму у повсякденній діяльності, а також при виконанні завдань з інших навчальних предметів: української мови, математики, трудового навчання тощо. Розширення кругозору учнів полягає в розумінні того, що самі діти, їх рідні, домашні тварини, технічні пристрої для вирішення власних потреб, задач, послуг та ін. щоразу виконують алгоритми. Формування міжпредметних компетентностей передбачає уміння виконання завдання з іншого предмету на основі алгоритмічного підходу, що ілюструє схема:

Завдання → Алгоритм → Результат.

Вироблення названих умінь за методичним підходом Г. Ломаковської та ін. потребує:

- актуалізації теоретичних знань (правила, означення, способу дій) з іншого навчального предмету;
- розгляд прикладу алгоритму (самостійне чи фронтальне його складання) на застосування теоретичних знань до розв'язування завдань;
- виконання алгоритму для 2-3 завдань.

**Висновки і перспективи подальших розвідок у даному напрямі.** Отже, аналіз чинних підручників та методичних посібників з вивчення курсу "Інформатика" у початковій школі дозволив розкрити методичні особливості вивчення основних понять теми "Алгоритми і виконавці": команда, виконавець, система команд виконавця, алгоритм. Зокрема, охарактеризовані суть та правила словесного способу подання алгоритмів, виділені відмінності методичних підходів щодо послідовності, прийомів та методів введення й формування понять, з'ясовані труднощі, що виникають у процесі набуття уміння складати алгоритми та розкриті методичні прийоми щодо їх усунення. До подальших досліджень у даному напрямі відносимо аналіз та узагальнення сучасних методичних підходів щодо реалізації інших тем курсу "Інформатика" у початковій школі.

#### **Використані джерела**

1. Інструктивно-методичні рекомендації щодо викладання інформатики у 2013-2014 н.р. [Електронний ресурс]. – Режим доступу: <http://mmk.edu.vn.ua/index.php/2011-10-04-12-01-08>
2. Конспекти уроків з інформатики 4 клас [Електронний ресурс]. – Режим доступу: <http://www.docme.ru/doc/76866/konspekti-urok%D1%96v-%D1%96nformatiki-4-klas>.
3. Державний стандарт початкової загальної освіти [Електронний ресурс]. – Режим доступу: <http://www.mon.gov.ua/education/average>

4. Земляк І.В. Методичні рекомендації щодо викладання інформатики в 3-му класі: за програмою "Сходинки до інформатики" / І.В. Земляк – Кам'янець-Подільський, 2009. – 114 с.
5. Коршунова О. В. Сходинки до інформатики : підруч. для 3-го кл. загальноосвіт. навч. закл. / О. В. Коршунова. – К.: Генеза, 2014. – 176 с. : іл.
6. Коршунова О.В. Методика викладання інформатики у 2-му класі/ О. В. Коршунова. – Х.: ФОП Співак В.Л., 2013. – 112 с.
7. Коршунова О.В. Сходинки до інформатики: підруч. для 2 кл. загальноосвіт. навч. закл. / О.В. Коршунова. – К.: Генеза, 2012. – 112 с.
8. Ломаковська Г.В. Сходинки до інформатики : підруч. для 2 кл. загальноосвіт. навч. закладів / Г.В. Ломаковська, Г.О. Проценко, Й.Я. Ривкінд, Ф.М. Рівкінд. – К.: Видавничий дім "Освіта", 2012. – 160 с.
9. Ломаковська Г.В. Сходинки до інформатики: підруч. для 3 кл. загальноосвіт. навч. закладів / Г.В. Ломаковська, Г.О. Проценко, Й.Я. Ривкінд, Ф.М. Рівкінд. – К.: Видавничий дім "Освіта", 2013. – 160 с.
10. Морзе Н.В. Методика навчання інформатики: Навч. посібник: у 4 ч./ за ред. акад. М.І. Жалдака / Н.В. Морзе. – К.: Навчальна книга, 2004. Ч. IV: Методика навчання алгоритмізації та програмування. – 368 с.
11. Сходинки до інформатики: Підруч. для 4 кл. загальноосвіт. навч. закл. / С.Я. Колесніков, Г.В. Ломаковська, Ф.М. Рівкінд, Й.Я. Ривкінд – 3 вид., перероб. – К.: Світич, 2009 – 69 с.
12. Програма курсу "Сходинки до інформатики": 2-4 класи загальноосвітніх навчальних закладів // Інформатика та інформаційні тех. в навч. закладах. – 2011. – №4-5. – С. 38-49.
13. Коршунова О. Зошит "Сходинки до інформатики" : навч. посіб. для загальноосвіт. навч. закл.: 2-й кл. / О.В. Коршунова. – К.: Генеза, 2013. – 64 с.

*Striletska N.*

**TO ISSUE OF STUDING THE THEME "ALGORITHMES AND DOERS"  
IN THE COURSE OF THE "INFORMATION" TECHNOLOGY  
FOR ELEMENTARY SCHOOL**

*An ensuring with exhaustive understanding of the methodological ideas of the initial course "An Information Technology" and taking advantage the methods and instruments of modern information technologies by students is one of the main assignments for the discipline "Methods of the studies of information technology" for the faculty of the elementary studies at the pedagogical higher educational institutions. Thus the problem of researching modern methodological approaches to realization of educational maintenance of course "Information Technology" for elementary school acquires actual character.*

*On the basis of analysis of valid textbooks from the course "Information Technology" for the elementary schools the author ascertained that the modern textbooks "The Steps to Information Technology" for second-third classes have a different approaches concerning introduction of concepts and acquaintance with the methods of algorithms' presenting. At the article were described the essence and rules of verbal algorithms' presenting, the differences of the methodical approaches of consistence, ways and methods of introduction and formation such concepts as "command", "doer", "doer's system of the commands", "algorithm", also were defined the troubles which arise in the process of acquiring of the ability to compose algorithms s and were described methodical ways of their removal.*

**Key words:** *algorithm, methods of algorithms' presenting, command, doer's environment, troubles of formation of knowledges about algorithm.*

*Стаття надійшла до редакції 02.09.2014 р.*