

Методика вивчення алгоритмів в початкових класах

Слово «алгоритм» походить від імені видатного узбецького математика Мухаммеда бена Муса аль-Хорезмі, який вперше описав десяткову систему числення і правила виконання арифметичних дій. Вважається, що саме частина імені вченого «аль-Хорезмі», що буквально означає «із Хорезма» (тобто із місцевості давнього Узбекистана), згодом трансформувалася у слово «алгоритм».

У Вікіпедії алгоритм визначається як набір інструкцій, які описують порядок дій виконавця, щоб досягти результату розв'язання задачі за скінченну кількість дій; система правил виконання дискретного процесу, яка досягає поставленої мети за скінченний час[1].

Алгоритм – це припис, який дозволяє виконати певну послідовність дій для досягнення результату. Властивості алгоритмів:

- зрозумілість, тобто розуміння виконавцем кожного кроку алгоритма;
- дискретність, тобто розбиття алгоритма на множину окремих скінчених дій (кроків);
- точність, тобто однозначність виконання припису;
- результативність, тобто обов'язкове отримання результату за скінчену кількість кроків;
- масовість, тобто можливість застосування даного алгоритма до цілого класу задач одного типу.

Формами представлення алгоритмів є вербальна (словесна), графічна (схеми, блок-схеми), мова програмування.

Звернення до теми «Алгоритми та виконавці» зумовлено об'єктивними причинами. Оновлення програми з інформатики для учнів початкових класів,

з одного боку, і використання відповідних засобів навчання, які вийшли у світ до 2017 року, з іншого, вносять незручності у роботу вчителя, особливо початківця. Це спонукало нас у межах цієї статті представити логіку подання навчального матеріалу, пов'язаного з алгоритмами з деякими методичними рекомендаціями.

Вивчення теми починається з актуалізації знань учнів щодо наявності та виконання ними у повсякденному житті різних команд, правил, приписів і т.ін. Наприклад, вчитель пропонує одному з учнів провести фізкультхвилинку, яка складається з чотирьох вправ (руки підняти догори, опустити їх, повернутися вліво, потім вправо). У процесі виконання завдання виникають питання: яким чином треба повертатися, всім тулубом, лише головою або повертатися від талії і т.ін. Організація фізкультхвилинки дозволяє окреслити проблемну ситуацію: чому діти по-різному виконували досить легке завдання. Причина криється у правильності формулювання команди, вірніше, у виборі такої її форми, яка б була однозначно зрозумілою іншими дітьми.

Таким чином, пояснюється, що команда – це речення, що спонукає до дії і є зрозумілим для того, хто її виконує. Для усвідомлення того, що не кожне спонукальне речення є командою, можна запропонувати завдання «Вибери із наведених речень ті, які є командами»:

- Запишіть домашнє завдання.
- Новий рік – родинне свято.
- Відкрийте підручник на сторінці 10.
- Роман правильно написав слово.
- Марійка хворіла на минулому тижні.
- Покладіть олівці на парти.

Команди втілюють у дію виконавці. Хто може бути виконавцем? Той, для кого команда є зрозумілою і можливою для виконання. Виконавцем команди може бути людина, тварина, технічний пристрій, комп'ютер тощо.

Так, доцільно зупинитися на визначенні різних виконавців. Можна запропонувати навести приклади (або продовжити наведений перелік) виконавців вдома, в школі, виконавців- тварин, виконавців-машин і т.ін.

- Вдома виконуються команди: помий посуд, прибери постіль, збери речі у портфель, вимкни телевізор, відчини двері, нагодуй рибок, тощо.
- У школі: запишіть речення, розв'яжіть задачу, принеси крейду, покладіть ручки на стіл тощо.
- Для тварин: Сидіти! Голос! Поруч!
- Для машин: натискання відповідних кнопок.

Проблемна ситуація: яким чином можна подати команду, крім спонукального речення?

- Жестом (суддя на спортивному майданчику, регулювальник на дорозі, диригент і т.ін.).
- Звуковим сигналом (свисток судді, дзвінок в театрі, що вказує на початок вистави, шкільний дзвінок, сигнал на пішохідному переході для незрячих людей і т.ін.).
- Світловим сигналом (світлофор, маяк, сигнали провідників на залізниці).

Для закріплення понять «команда» і «виконавець» можна запропонувати скласти команди із поданих словосполучень та визначити можливих виконавців:

- мити руки;
- їсти;
- збирати речі;
- йти на прогулянку;
- прочитати текст;
- принести м'ячик.

Цікавим методичним прийомом є визначення у дитячих піснях або віршах виконавців та команд, які вони мають виконати. Наприклад, Чуркіна Т.І. пропонує виконати таку роботу з текстом, наведеним на рис. 1[4].

**Визнач виконавців та команди, які
вони мають виконати.**

Ой ти, коте сірий,
та вимети сіни.
А ти, волохатий,
прибери у хаті.
А ти, коте чорний,
та сідай у човен.
Пливи, котку, на лужок
забавляти діточок.



Рис. 1.-Слайд презентації «Команди та їх виконавці»

Наступним етапом є пояснення про послідовність дій, як ланцюга із декількох команд. Це доцільно робити, спираючись на життєвий досвід учнів. Так, на побутовому рівні послідовністю дій є приготування їжі або напою за рецептом (для приготування чаю: налити воду в чайник, включити чайник, взяти чашку, покласти пакетик чаю, залити окропом, додати цукру); виконання вранішніх гігієнічних процедур (включити воду, взяти зубну щітку, вичавити зубну пасту, почистити зуби, умитися, виключити воду, обтертися рушником).

Паралельно наводимо приклади з природи, наприклад, щодо пророщення фасолі (взяти посуд, покласти кульку з вати, налити води, покласти фасоліну, поставити на підвіконня, чекати пару днів); пересадка рослини і т.ін.

В ігровій формі можна показати до чого призводить зміна послідовності дій і аналіз того, який результат отримується в залежності від такої зміни. Наприклад, на окремих листах записані команди:

1. Підійти до дошки.
2. Взяти крейду.
3. Намалювати сонечко.
4. Взяти ганчірку.
5. Витерти дошку.

6. Сісти на місце.

До участі у грі запрошується 3-4 учні, кожному з яких необхідно виконати всі 6 команд, але їх послідовність визначається простим перемішуванням листків. Таким чином, у кожного гравця буде своя послідовність дій, яка призводить до різних результатів. Разом з дітьми підводимо до висновку, що виконавець отримає бажаний результат, якщо команди будуть зрозумілими, під силу виконавцю і розташовані у логічній послідовності.

Для пропедевтики визначення комп'ютера як електронного виконавця доцільно познайомити учнів з Роботом, механічним виконавцем команд. Згадавши про героїв фільмів, оповідань, можна організувати бесіду щодо наявності роботів у повсякденному житті (пральна, посудомийна машини, мультиварка тощо) і правильності команд для їх функціонування. Закріпленням може стати відома гра «Так або ні», в якій відповідати на запитання можна тільки «так» або «ні»:

- Роботи – це жива істота?
- Роботи допомагають людям?
- Роботи відповідають на питання?
- Роботи можуть співати?
- Робот зрозуміє команду «Іж!»?
- Послідовність команд для Робота може містити одну команду?
- Послідовність команд для Робота може містити десять команд?
- Робот може сам собою керувати? І т.ін.

Прикладом виконання послідовності дій, що призводить до предметного результату є виконання графічних диктантів, які є корисними також для формування просторової орієнтації, графічних та обчислювальних навичок учнів [3]. Наприклад, виконання такої послідовності команд призведе до намалювання песика(рис.2):

2↑ 4→ 2↓ 1→ 4↑ 1→ 1↑ 1← 1↓
 5← 3↑ 1→ 1↑ 1← 1↓ 3← 2↓ 2→
 5↓ 1→

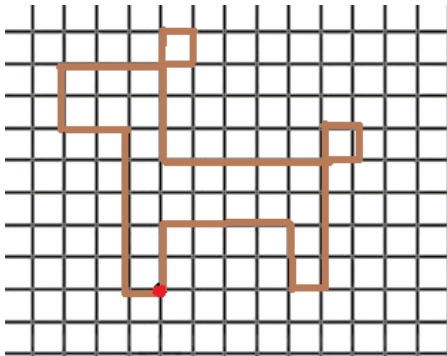


Рис.2.-Результат графічного диктанту «Песик»

Програмою передбачено формування у молодших школярів вмінь наводити приклади послідовних дій в побуті; знаходити повторюваність подій в казках, повторювані стани у природі; знаходити помилки в поданих записах послідовних дій і т.ін. Так, прикладом повторюваності подій в казках є слова матері Івасика-Телесика:

Телесику, Телесику!
 Приплинь, приплинь до бережка!
 Дам я тобі їсти й пити!

У тій же казці Телесик декілька разів промовляє до гусей з проханням віднести його додому:

Гуси-гуси, гусенята!
 Візьміть мене на крилята
 Та понесіть до батенька,
 А в батенька їсти, й пити,
 Ще й хороше походити!

У природі повтори можуть бути сезонними (зима, весна, літо, осінь); часовими (доба, тиждень, місяць і т.ін.); у житті різних тварин (зимова сплячка, переліт у вирій і т.ін.). Подібні приклади дозволяють активізувати

пізнавальну діяльність учнів та показати міжпредметний характер теми вивчення, а також в подальшому стати підґрунтям для пояснення алгоритмів з повторенням та умовами.

Унаочненням може бути робота за комп'ютером з виконавцями в середовищах програмування, зокрема з такими, як «Навантажувач», «Садівник», «Кенгуру» («Сходи́нки до інформатики»).

Робота з комп'ютерною програмою «Навантажувач» передбачає формування вмінь в учнів правильно давати команди, у даному випадку для завантаження корабля контейнерами. Кількість контейнерів може бути різною. Контейнери можна розміщувати на кормі (задня частина корабля) і на носі (передня частина корабля) так, щоб вага в обох частинах корабля була однаковою. За допомогою визначених команд учням необхідно правильно завантажити корабель, а потім і відшвартувати корабель для того, щоб той вийшов з порту у море.

Робота з цією програмою унаочнює безпосереднє виконання команд, послідовності команд, помилкові команди і т.ін.

Наступний етап вивчення теми - це визначення плану дій, який дозволяє послідовність дій або команд побудувати як інструкцію для виконання того чи іншого задуму. Доцільною є бесіда на цю тему:

1. Чи знаєте ви, що таке план?
2. Чи планували ви свою діяльність?
3. Чи планують якісь дій ваші батьки?
4. Як ви вважаєте планування допомагає у вирішенні завдань чи навпаки, їх заплутує? Поясніть свою відповідь.
5. На уроках математики ми розробляємо план розв'язування задачі. Згадайте, з якою метою ми це робимо?
6. Перед виходом на екскурсію вчитель повідомляє учням план цього заходу. Як ви вважаєте, з якою метою?

7. Як ви розумієте вислів «Твої плани на майбутнє (вихідні, канікули тощо)»?

Таким чином, підходимо до розуміння того, що план – це заздалегідь розроблена послідовність дій, яка призводить до очікуваного результату. Планування дозволяє раціонально використати час, вилучити із послідовності дій зайві, тобто такі, що не впливають на результат.

Наприклад, організація гри, в ході якої необхідно розташувати всі малюнки таким чином, щоб отримався план дій під назвою «Розпорядок дня»(рис.3):



Рис. 3.- Слайд із презентації «Алгоритми і виконавці»[2]

У ході гри обговорюються різні варіанти розташування малюнків з метою з'ясування правильного плану дій. Інколи план дій називають інструкцією. Інструкція - це вказівка на порядок і засоби виконання якої-небудь роботи, користування приладами, інструментами. Призначення інструкцій дуже різне, вони можуть вказувати як користуватися різними приладами, інструментами, як правильно грати в різні ігри, як виготовити власноруч м'яку іграшку чи виконати конкретну роботу.

Плани та інструкції розробляються спеціально для виконавців. Цікавим фрагментом на уроці може стати показ популярного мультфільму

про Фіксиків, зокрема серія «Інструкція», в якій герої розповідають про те, що інструкції дають корисні поради: в інструкції про шафу є схема її збору, в інструкції до телевізора є вказівки про налаштування програм, в інструкції до каші – рецепт її приготування, в інструкції до ліків – поради щодо їх правильного вживання [7]. Так, головний герой був впевнений, що складе стільчик без інструкції. Але перевірка на міцність показала, що стільчик не витримав навантаження, оскільки були використані не всі деталі. Впродовж можна запропонувати такі завдання:

1. Складіть інструкцію для виготовлення літака з паперу. Хто може бути виконавцем цієї інструкції? Скільки команд містить ваша інструкція?
2. Складіть інструкцію, яка допомагає додати два двоцифрових числа у стовпчик без переходу через розряд (з переходом через десяток).
3. Складіть інструкцію для приготування бутерброду. Що впливає на кількість команд? Чи може приготувати бутерброд виконавець, який не знає про очікувані результати?

Так, підходимо до визначення алгоритму, як скінченної послідовності інструкцій (команд), виконання яких приводить до правильного розв'язання поставлених задач і отримання певного результату.

Формальне виконання алгоритму передбачає таке його виконання, коли сам виконавець не знає ні постановки задачі, ні змісту одержаних результатів, але, чітко виконуючи усі дії, записані в алгоритмі, досягає необхідного результату. На даному етапі найбільш вдалим прийомом є організація рольових ігор, в ході яких гравці поділяються на виконавців та на інструкторів. Перші виконують команди других, не задумуючись про кінцевий результат. Виконавець повинен вгадати, що він робить і дати назву алгоритму. Наприклад, виконавцю № 1 подають команди:

1. Візьми чистий аркуш в клітинку.

2. Візьми олівець.
3. Відступи 4 клітинки вправо, постав точку.
4. Приклади лінійку, відміряй 10 см, постав точку.
5. З'єднай дві точки прямою лінією.

Висновок: алгоритм креслення відрізка довжиною 10 см.

Виконавцю № 2 подають команди:

1. Задумай число.
2. Помнож це число на 2.
3. Додай до результату 8.
4. Поділи отримане число на 2.
5. Відними від результату задумане число.
6. Запиши результат.

Висновок: алгоритм отримання числа 4.

До речі, другий алгоритм відноситься до так званих математичних фокусів, коли за допомогою проведення нескладних арифметичних операцій «ведучий» вгадує число або дату.

Одним із найвідоміших таких «фокусів» є алгоритм відгадування дня народження людини. У цьому випадку є алгоритм дій для того, хто відгадує дату, і для того, хто бажає пересвідчитися у тому, що дату його народження вгадають правильно. Для гравця:

1. Помножте на 2 число дня свого народження.
2. Додайте число 5.
3. Помножте на 50 отриману суму.
4. Додайте до результату номер місяця народження (наприклад, лютий – це 2, вересень- 9).
5. Назвіть отримане число.

«Фокусник» після отримання числа, віднімає від нього 250. Має отриматися чотирицифрове число. Якщо отримується трицифрове,

дописується на початку числа 0. Перші дві цифри числа, що отрималося-день народження, дві останні- місяць.

Методично доцільними на даному етапі вивчення теми є завдання, в яких необхідно визначити пропущену команду в запропонованому алгоритмі. Наочно це можна зробити за допомогою виконавця «Садівник» з програмного засобу «Сходи до інформатики», робота з яким передбачає як покрокове, так і повне виконання команд. В обох випадках пропуск однієї з команд унеможлиблює виконання наступної дії.

Вчитель показує, що алгоритмами учні користуються з перших днів навчання у школі: правила письма, креслення, виконання арифметичних дій на письмі, організація та проведення дослідів, різноманітні вироби на уроках трудового навчання тощо. У таких випадках алгоритм являє собою корисний інструмент для вирішення завдань з різних галузей знань або побуту: математики, рідної чи іноземної мови, природознавства, конструювання, інформатики або повсякденного життя.

Ускладненим варіантом є завдання розташувати команди алгоритму у правильній послідовності. Наприклад, алгоритм пришивання гудзика до штанів:

1. Відрізати нитку.
2. Покласти голку та ножиці на місце.
3. Вдіти нитку у голку.
4. Прикласти гудзик до потрібного місця штанів.
5. Зав'язати вузлик.
6. Протягнути голку з ниткою через вушко гудзика та тканину декілька разів.
7. Взяти відрізок нитки потрібного кольору.
8. Зав'язати вузлик.

Для формування навичок алгоритмізації в якості навчального матеріалу можна використовувати найрізноманітніші види завдань. Наприклад, сортування та аналіз даних, пошук потрібної інформації, табличне або графічне представлення даних, розв'язання задач різних видів і т.ін.

Наведемо декілька прикладів алгоритмів, які використовуються у початкових класах, а їх складання сприяє формуванню міжпредметних компетентностей у молодших школярів.

1. Послідовність розбору слова за будовою:

- визначити слово за питаннями і виділити закінчення;
- назвати основу слова;
- добрати до слова два-три споріднені та визначити корінь;
- визначити префікс (якщо є);
- визначити суфікс (якщо є).

2. Написання ненаголошених [e], [u] у корені слова:

- взяти слово;
- змінити слово або дібрати до нього спільнокореневі так, щоб ненаголошений голосний став наголошеним;
- писати той голосний, який опинився під наголосом;
- інакше перевірити за словником.

3. Визначення відміни іменника вереднього роду:

- взяти іменник середнього роду;
- поставити іменник у називному відмінку;
- якщо іменник закінчується на -о,-е,-я, то іменник другої відміни;
- якщо іменник закінчується на -ат,-ят,-ен, то іменник четвертої відміни.

4. Знаходження периметра прямокутника:

- знайти суму двох суміжних сторін прямокутника;

- отриману суму помножити на 2.

5. *Алгоритм додавання (віднімання) способом округлення (Якщо запис одного з доданків або від'ємника закінчується однією з цифр 5, 6, 7, 8 або 9):*

- замінити це число близьким круглим числом;
- додати (відняти) це кругле число;
- визначити, на скільки одиниць збільшили до круглого числа;
- додати (відняти) цю кількість одиниць;
- записати відповідь.

6. *Алгоритм заповнення календаря природи:*

- визначити стан неба (ясно, хмарно, похмуро);
- записати опади (сніг, дощ, іній, туман);
- подивитися температуру;
- визначити по деревах напрям і силу вітру та позначити стрілкою.

Наступним етапом у вивченні алгоритмів є пояснення способів їх представлення та видів. Доцільною є організація бесіди, під час якої вчитель разом з учнями з'ясовує, яким чином, у якому вигляді виконавець отримує алгоритм дій.

- Діти, давайте згадаємо правило написання великої літери у слові.
- Чи можна це правило назвати алгоритмом? Чому?
- З яких джерел ви дізналися про цей алгоритм?
- Так, це правило є алгоритмом, оскільки його виконання дозволяє за певну кількість кроків отримати правильну відповідь і визначитися з написанням літери. А дізналися ви про це зі слів вчителя та з підручника. У таких випадках алгоритм представлено у словесній або текстовій формі.
- На рисунку 4 зображено алгоритм відгадування дати народження, про який йшлося вище. Цей алгоритм представлено у графічній формі.

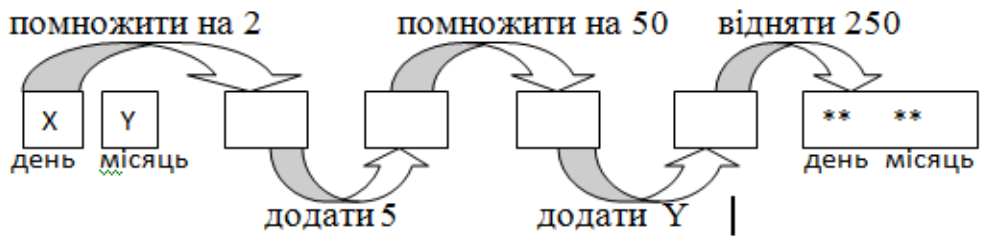
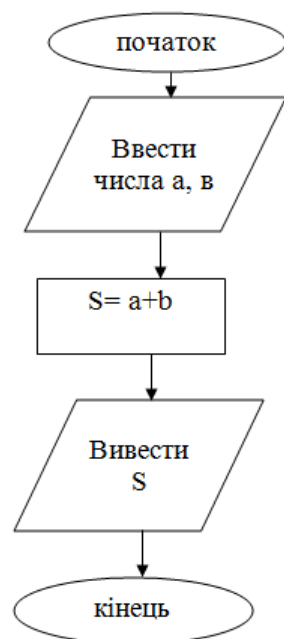


Рис. 4. –Алгоритм відгадування дня народження

Графічний спосіб є зручним і зрозумілим. В інформатиці для графічного зображення алгоритмів прийнято використовувати блок-схеми.

Кожний блок має своє загально визначене призначення, зрозуміле людям будь-якої національності: блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних — прямокутником, блок прийняття рішень — ромбом, еліпсом — початок та кінець алгоритму.

Так, наприклад блок-схема алгоритму знаходження суми двох чисел має вигляд:



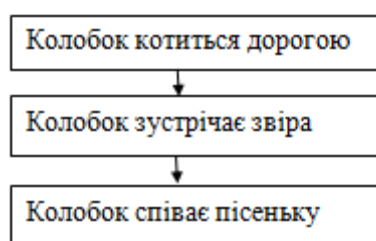
Саме за допомогою блок-схеми легко пояснити дітям, чому подібні алгоритми називаються лінійними (всі дії виконуються послідовно, одна за одною, наче по лінійці). Далі можна запропонувати учням скласти блок-схеми для різноманітних лінійних алгоритмів («Зліпити Сніговика», «Приготувати бутерброд», «Відправити СМС-повідомлення» і т.ін.).

Введення понять циклічного та розгалуженого алгоритмів легко сприймаються дітьми за допомогою відомих казок. Так, наприклад, звернення до казки «Колобок» дозволяє вилучити інформаційний блок, що повторюється декілька разів: Колобок котиться доріжкою, зустрічає різних звірів та співає їм пісеньку:

Я по засіку метений,
Я із борошна спечений,—
Я від баби втік,
Я від діда втік,
То й від тебе втечу!

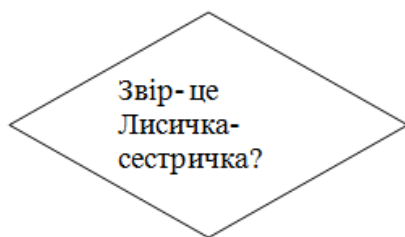
Як відомо, казка закінчується, як тільки Колобок зустрічає Лисичку-сестричку.

Пояснюємо, що алгоритм, в якому дія повторюється декілька разів називається *циклічним*. Повторюваність закінчується як тільки буде виконано умову. З'ясуємо, який саме блок повторюється:

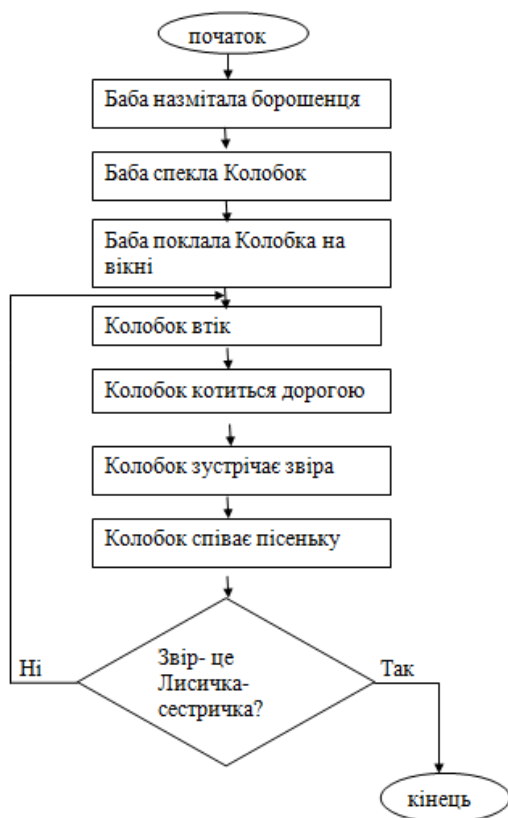


А умовою є зустріч з Лисичкою-сестричкою. Якщо умова не виконана, Колобок зустрічає наступного звіра, співає йому пісеньку та котиться далі.

Якщо ж умова виконується, то алгоритм закінчується. Умову позначаємо на блок-схемі у вигляді ромба:



Спробуємо об'єднати всі блоки в єдину схему:



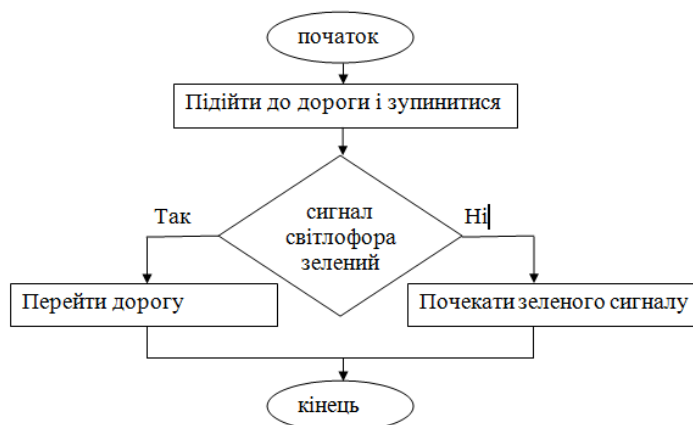
Пропонуємо учням згадати казки, в яких також дії повторюються, поки не виконається казкова умова. Такими казками є «Рукавичка», «Івасик-Телесик» та ін. Алгоритм можна представити як у текстовій формі, так і у графічній.

Пояснюємо учням, що відмінною ознакою циклічного алгоритму є повторюваність виконання дій, **поки** не буде виконана умова.

З метою визначення розгалуженого алгоритму акцентуємо увагу учнів на блоці прийняття рішень, тобто на умові, від якої залежить або повторення циклу або його завершення.

- Якою геометричною формою позначено на блок-схемі умову? (Ромб)
- А чи можете ви назвати випадки з власного життя, коли від вибору умови залежить наступна дія?
- Робота світлофора впливає на наш вибір переходити дорогу чи зачекати, шкільний дзвоник дозволяє нам виходити на перерву або заходити в клас, перевірка правилом допомагає вірно написати слово і т.ін.

У першому випадку блок-схема матиме вигляд:



Таким чином, алгоритм, в якому є вибір дії в залежності від виконання умови, називається *розгалуженим*.

- Згадайте, чи є казкові умови, від вибору виконання яких залежить подальша доля героїв? Спробуйте оформити це у вигляді блок-схеми. (Наліво підеш – коня загубиш, направо – голову втратиш).

Акцентуємо увагу учнів на відмінній ознаці розгалуженого алгоритму: **якщо** умова вірна, **то** виконується правильна дія. З цією метою доцільно

навести приклади з різних дисциплін з промовлянням саме конструкції **якщо-то**:

1. **Якщо** число ділиться на 2, **то** воно парне.
2. **Якщо** квітку не поливати, **то** вона зів'яне.
3. **Якщо** іменник у реченні відповідає на питання хто?або що?, **то** він є підметом.

Більшість прислів'їв можна перефразувати у вислів, побудований за конструкцією **якщо-то**. Це дає можливість використання завдань, в яких треба виконати такий перефраз. Наприклад, прислів'я «хто багато читає, той багато знає» перефразуємо в таке: «**якщо** багато читати, **то** будеш багато знати». Корисною є й обернена справа: речення «**якщо** сила не візьме, **то** розум допоможе» є перефразуванням відомого прислів'я: «де сила не візьме, розум допоможе».

Як зазначалося вище, функціональність алгоритмів доцільно перевіряти у відповідних середовищах програмування. У засобі «Сходинки до інформатики» представлено виконавців «Навантажувач», «Садівник», «Кенгуру». Частково ми зупинялися на їх використанні на етапі первинного ознайомлення з алгоритмами і для створення лінійних. Але й під час створення циклічних та розгалужених алгоритмів вони можуть стати корисними для «маленьких програмістів». Так, для Садівника передбачені команди «повтори...разів» та «повтори поки є саджанці». Правильне вживання цих команд у записі програми унаочнюють виконання циклічних алгоритмів.

Аналогічно, виконавець «Кенгуру» дозволяє вводити у програму різні повтори. Цікавими є завдання на створення у цьому середовищі алгоритмів малювання різноманітних орнаментів або лабіринтів.

Після того, як учні опанують вміннями створювати циклічні алгоритми за допомогою зазначених виконавців, рекомендуємо знайомство з середовищем програмування для дітей Scratch.

Скретч - інтерпретована візуальна мова програмування. Завдяки динамічності, вона дає змогу змінювати код навіть під час виконання. Мова має за мету навчити дітей поняттю програмування і дає змогу створювати ігри, анімації чи музику. Scratch — середовище програмування, яке дозволяє дітям створювати власні анімовані та інтерактивні історії, ігри й інші проекти. Ними можна обмінюватися в мережі Інтернет.

Це об'єктно-орієнтоване середовище, в якому блоки програм збираються з різнокольорових «цеглинок» команд так само як машини збираються з різнокольорових блоків у конструкторах Лего.

У результаті виконання простих команд створюється складна модель, в якій взаємодітимуть багато об'єктів, наділених різними властивостями[6].

Таким чином, запропонована логіка подачі навчального матеріалу дозволяє вчителю побачити не тільки етапи основ алгоритмізації, але й створити власну авторську концепцію розкриття теми «Алгоритми та виконавці», незалежно від того, які підручники, робочі зошити чи посібники є в наявності. З огляду на обмеженість об'єму статті, подальшого дослідження, на наш погляд, потребує більш детальне вивчення можливостей українських народних казок та прислів'їв як змістового наповнення алгоритмів, що сприятиме формуванню предметних і міжпредметних компетентностей учнів початкових класів.

Використані джерела:

1. Алгоритм:[Електронний ресурс].- Режим доступу:
<https://uk.wikipedia.org/wiki/>

2. Алгоритми і виконавці:[Електронний ресурс].- Режим доступу:
<http://svitppt.com.ua/informatika/algoritmi-i-vikonavci-algoritmu-sposobi-zapisu-algoritmu-liniyniy-algoritm.html>
3. Графічні диктанти рідною мовою:[Електронний ресурс].- Режим доступу:
<http://ridna-mova.com.ua/63-grafchn-diktanti.html>
4. Команди та їх виконавці. Презентація вчителя початкових класів Черкаської загальноосвітньої школи I-III ступенів № 29 Чуркіної Тетяни Іванівни:[Електронний ресурс].- Режим доступу:
<http://svitppt.com.ua/informatika/komandi-ta-ih-vikonavci1.html>
5. Матеріали для закріплення, проведення самостійних і контролюючих робіт по розділу «Алгоритмика» для молодших школярів:[Електронний ресурс].- Режим доступу: <http://www.metod-kopilka.ru/page-test-8-5-12.html>
6. Скретч (мова програмування):[Електронний ресурс].- Режим доступу:
<https://uk.wikipedia.org/wiki>
7. Фіксикі-Інструкція:[Електронний ресурс].- Режим доступу:
https://www.youtube.com/watch?v=ET838We_UvE