**Володимир БУНЯК,**

*методист кабінету математики*

*та технологій Рівненського ОІППО*

**Андрій ВИБАЧ,**

*головний спеціаліст управління*

*освіти і науки Рівненської ОДА*

**Методичні рекомендації щодо навчання інформатики**

**У 2020–2021 навчальному році**

У 2020–2021 навчальному році вивчення інформатики в основній та старшій школі закладів загальної середньої освіти здійснюватиметься за навчальними програмами, які розміщено на офіційному веб-сайті Міністерства освіти і науки України (*див. таблицю*):

**Навчальні програми для вивчення інформатики в основній**

**та старшій школі закладів загальної середньої освіти**

|  |  |  |
| --- | --- | --- |
| **Класи****(рівні)** | **Рік затвердження програми** | **Посилання** |
| ***Основна школа (5–9 класи)*** |
| 5–9 | 2017 | <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-5-9-klas/onovlennya-12-2017/8-informatika.docx> |
| ***Поглиблене вивчення інформатики*** |
| 8–9 | 2016 | <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-5-9-klas/informatika.pdf> |
| ***Старша школа (10-11 класи)*** |
| Рівень стандарту | 2017 | <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-10-11-klas/2018-2019/informatika-standart-10-11.docx> |
| Профільний рівень | <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-10-11-klas/2018-2019/01/10-11-profilniy-riven.docx> |

***Основна школа***

У 2019–2020 навчальному році за новою навчальною програмою з інформатики, призначеною для учнів, що вивчали інформатику у 2–4 класах, вперше навчатимуться учні ***9 класів***. Це є продовження так званого другого рівня (8-9 класи) за новою програмою для 5-9 класів (перший рівень був для 5-7 класів). Другий рівень передбачає завершення повноцінного формування ключових та предметних ІТ-компетентностей. На цьому рівні, зокрема, має формуватися понятійний апарат, достатній для набуття відомих 10 ключових компетентностей в курсі інформатики. Для цього рекомендується використовувати повнофункціональні, а не імітаційні, програмні засоби та середовища.

Очікувані результати навчання вказано у змістовому розділі програми для кожної теми курсу в кожному класі. Час, що необхідний для досягнення цих результатів, визначається вчителем залежно від рівня попередньої підготовки учнів, обраної методики навчання, наявного обладнання тощо. Однак на опанування тем змістової лінії «Моделювання, алгоритмізація та програмування» має приділятися не менше 40 % навчального часу в 5–8 класах і не менше 30 % у 9 класі. За необхідності вчитель може змінювати порядок вивчення тем, не порушуючи змістових зв’язків між ними. Отже, у новому навчальному році згідно нової програми, учні ***9 класів*** будуть розглядати п’ять тем:

* Програмне забезпечення та інформаційна безпека
* 3D-графіка
* Опрацювання табличних даних
* Бази даних. Системи керування базами даних
* Алгоритми та програми

***Рекомендації щодо навчання учнів основам алгоритмізації та програмування***

***у базовому курсі інформатики***

Сьогодні навчальні заклади України на шляху до кардинальних змін. Не є виключенням і предмет «Інформатика» та найважливіший його розділ «Основи алгоритмізації та програмування». Оволодівши необхідними знаннями та навичками в області алгоритмізації та програмування, у людини з’являється можливість створювати власні та вдосконалювати існуючі доробки у сфері інформаційних технологій. Виходячи з означення інноваційного потенціалу, алгоритмізація та програмування може використовуватись у навчально-виховному процесі як один із засобів для підвищення інноваційного потенціалу особистості як вчителя, так і учня.

Під час вивчення інформатики в учнів має розвиватися алгоритмічне, структурне, логічне, критичне мислення, а також творчі здібності. В той час як розвиток логічного і критичного мислення, а також творчих здібностей не є завданням винятково курсу інформатики, а відбувається не меншою мірою під час вивчення інших дисциплін, розвитку алгоритмічного та структурного мислення на уроках інформатики слід приділяти першочергове значення.

Алгоритмічне мислення - це вміння:

• визначати послідовність дій, які необхідно виконати для розв'язання певної задачі;

• подавати алгоритми в певному формальному вигляді і виконувати їх;

• використовувати в алгоритмах алгоритмічні структури, тобто розгалуження і повторення;

• опрацьовувати величини, сталі й змінні;

• обирати оптимальний алгоритм розв'язання задачі.

Структурне мислення - це вміння:

• визначати параметри об'єктів та їх можливі значення;

• класифікувати явища та об'єкти;

• знаходити структурні та ієрархічні зв'язки між класами об'єктів, класифікувати зв'язки;

• розв'язувати задачі з обробки структур даних (передусім - формалізовувати вимоги щодо відбору даних за певними критеріями);

• подавати дані в табличному та графічному вигляді, інтерпретувати дані, подані графічно.

Слід зазначити, що алгоритмічне мислення має формуватися не лише під час вивчення основ алгоритмізації та програмування, а й під час виконання навчальних проектів, розв'язання компетентнісних задач та вивчення інших тем курсу. Зокрема, послідовні алгоритми, алгоритми з розгалуженнями та робота з величинами мають закріплюватися задачами з опрацювання даних в окремих клітинках електронних таблиць, а опрацювання наборів числових даних в електронних таблицях може бути пропедевтикою вивчення алгоритмічної структури повторення.

Вчителю при вивченні теми «Основи алгоритмізації та програмування» в базовому курсі інформатики пропонується уникати використання штучних навчальних або застарілих середовищ програмування. Усі програми, які розроблятимуть учні, мають відповідати принципам функціонування сучасного програмного забезпечення в середовищі операційної системи з графічним інтерфейсом. Винятком є олімпіади з програмування.

За новою програмою вивчення алгоритмізації та програмування повертається у шкільний курс інформатики і розглядається у кожній паралелі, починаючи з другого класу.

Зауважу, що у даних рекомендаціях ми будемо говорити лише про методичні аспекти вивчення алгоритмізації та програмування у базовому курсі інформатики (5-9 класи).

Вивчення алгоритмізації у шкільній інформатиці може мати два цільових напрями:

* розвивальний напрям, під яким розуміється розвиток алгоритмічного мислення учнів;
* програмістський напрям, що передбачає в собі вивчення технології створення програм.

Останній напрям можна розділити на два цільових аспекти:

* перший аспект пов'язаний з посиленням фундаментальної компоненти курсу інформатики. Учням дається уявлення про те, що таке мови програмування, що представляє собою програма мовою програмування високого рівня, як створюється програма в середовищі сучасної системи програмування.
* другий аспект носить профорієнтаційний характер. Вивчення програмування в рамках шкільного курсу дозволяє учням випробувати свої здібності до такого роду діяльності і, при бажанні, вибрати у майбутньому відповідний професійний шлях.

Алгоритмізація у школі відповідає методу структурного програмування і є підготовчим етапом до вивчення об’єктно-орієнтованого програмування, актуального на сучасному етапі розвитку програмування. Стандартна програма вивчення основ алгоритмізації передбачає наступну послідовність тем:

* складання лінійних алгоритмів;
* складання циклічних алгоритмів;
* використання розгалужень в алгоритмах;
* опис і використання допоміжних алгоритмів.

На початковому етапі вивчення певної структури алгоритму доцільно використовувати блок-схеми, які наочно демонструють базові структури алгоритмів та дають можливість сформувати правильну уяву про механізм роботи кожної із них. ([Демонстрація](https://docs.google.com/presentation/d/1fRAaj-QAXSgCAgFkAjffiEO5Yr6H4uovQdw9tqzn86w/edit#slide=id.g17c080cda2_0_0))

**Примітка:** ця і всі наступні демонстрації надані лише для перегляду. Бажано для цього мати обліковий запис Google. Для бажаючих редагувати приклади в демонстраціях потрібно запитати дозволу в автора.

Ефективним засобом підвищення рівня сприйняття теорії алгоритмізації є використання середовищ з виконавцями, що працюють у певній обстановці. Такі виконавці наочно представляють механізм виконання алгоритмів та їх базових структур.

([Демонстрація](https://docs.google.com/presentation/d/1fRAaj-QAXSgCAgFkAjffiEO5Yr6H4uovQdw9tqzn86w/edit#slide=id.p3))

Програма курсу інформатики пропонує в якості такого середовища використовувати Скретч, але, власний досвід показав, що використовувати його як основний інструмент при вивченні алгоритмізації дещо складно і не завжди раціонально. Та й використовувати це середовище бажано лише до 6 класу включно.

На мою думку більш ефективним буде використання сайту **code.org**, який створений саме з навчальною метою. На сайті пропонуються завдання, розбиті на курси. Кожен курс розрахований на певний рівень підготовки учня та його вік, і розбитий на етапи, що містять завдання з окремо взятої теми.

Процес виконання завдань учнем системою контролюється і у разі неправильного, або нераціонального виконання завдання система реагує, пропонуючи учневі той чи інший наступний крок.

Вчитель може на цьому сайті створити групи, що дає йому можливість вказувати курс, який кожна група повинна пройти та слідкувати за результативністю виконання завдань кожним з учнів групи.

При вивченні алгоритмізації можна використовувати курси 1-4

    1. Скретч. Демонструються приклади створених проектів (наприклад: "Заїхати у гараж").

    2. Коде.орг. Демонструються можливості, які надає сайт для використання у навчально-виховному процесі. Курси, що пропонуються на сайті, приклади завдань з курсів 1,2,3,4:

            Курс 1 - Етап 13 - Завдання 5,

            Курс 2 - Етап 8 - Завдання 2,

            Курс 3 - Етап 13 - Завдання 5 (Розгалуження),

            Курс 4 - Етап 12 - Завдання 5 (Цикл-Поки)

**Додаткові поради щодо вивчення алгоритмізації та програмування (з власного досвіду):**

5 класи:

Коде.орг - Курс 2, Курс 1 - Етапи 12-18, Курс 3 - Етап 12.                       ([Демонстрація](https://docs.google.com/presentation/d/1fRAaj-QAXSgCAgFkAjffiEO5Yr6H4uovQdw9tqzn86w/edit#slide=id.g18d030c718_0_0))

Теми розділу:

* Лінійні алгоритми
* Структура повторення з лічильником
* Структура розгалуження
* Структура повторення з умовою

6 класи:

Коде.орг - Курс 1 - Етапи 1-11, Курс 2 - Етапи 1-4.

Скретч (коли відсутній Інтернет (самостійне опрацювання за підручником))

Теми розділу:

* Об’єкти та події
* Команди та виконавці
* Алгоритми та програми
* Лінійний алгоритм. Складання алгоритмів

7 класи:

Коде.орг - Курс 2, Курс 3 - Етапи 2, 3, 7, 12                   ([Демонстрація](https://docs.google.com/presentation/d/1fRAaj-QAXSgCAgFkAjffiEO5Yr6H4uovQdw9tqzn86w/edit#slide=id.g18d4f7d28e_0_13))

Скретч (коли відсутній Інтернет (самостійне опрацювання за підручником))

Теми розділу:

* Структура повторення з лічильником
* Висловлювання. Логічне слідування. Структура розгалуження
* Структура повторення з умовою

8-9 класи:

Для подальшого вивчення алгоритмізації та програмування під час практичної реалізації набутих знань рекомендується використовувати одну з мов програмування високого рівня (Pascal, C, C++, C#, Python, Java, Lazarus та ін.)

Варто зауважити, якщо в даних рекомендаціях не містяться поради щодо будь-якого виду навчальної діяльності (наприклад, робота з обдарованими учнями, позакласна робота тощо), дійсними лишаються рекомендації МОНУ та РОІППО, подані протягом останніх трьох років.

***Старша школа***

Змін не передбачається. Вивчення інформатики у старшій школі відбуватиметься за рекомендаціями МОНУ на 2019-2020 н. р. (для всіх рівнів). Але це може змінитися після публікації нових рекомендацій МОНУ.

***Організація сучасного уроку інформатики***

Методика проведення кожного уроку з інформатики визначається вчителем з урахуванням того, що обов’язковою передумовою успішного виконання вимог програми є практична діяльність учнів з індивідуальним доступом кожного учня до роботи з персональним комп’ютером.

При плануванні та підготовці до уроків учителю варто зважати на основні принципи шкільної інформатики:

1. Застосування на практиці отриманих знань та навичок, розвиток предметних та ключових компетентностей учнів.
2. Спрямованість на реальне життя та інтеграцію з іншими предметами.
3. Активне навчання та творчість.
4. Інновації як в освіті, так і в технологіях.
5. Спільна навчальна діяльність завдяки роботі в парах та малих групах.
6. Створення нових інформаційних продуктів та пошук нових знань.
7. Вільний вибір програмних засобів та онлайн-сервісів для навчальної та практичної діяльності, зокрема можливість використання вільно поширюваного програмного забезпечення як альтернативи пропрієтарним програмним продуктам.
8. Використання безпечних веб-середовищ та дотримання конфіденційності мережевої особистості учнів.
9. Дотримання авторських прав розробників програм, добропорядне використання контенту.

Важливим чинником розвитку ключових компетентностей є інтегрованість змісту уроку інформатики, яка передбачає:

* проблемну орієнтованість пропонованих на уроках завдань, що стимулює дискусію, обговорення, пошук різноманітних джерел інформації, зіткнення думок і переконань;
* пов’язаність змісту уроку з реальним життям;
* практичну значущість інформації, що знаходить підтвердження через реальні факти та в змодельованих на уроці ситуаціях.

Ключові компетентності можна розвивати завдяки відповідним формам роботи, які відображають комунікативно-діяльнісний підхід до навчального процесу.

Із цією метою потрібно використовувати:

* інтерактивні форми та методи роботи, які забезпечують активну діяльність учнів у процесі опанування навчального матеріалу;
* кооперативне навчання, під час якого формуються соціальні вміння, лідерські якості;
* рольові та ділові ігри, що допомагають пізнати світ і себе в ньому, підвищують самооцінку та попит на інновації;
* проектні технології, завдяки яким в учнів формується проектне мислення, почуття відповідальності та досвід цілеспрямованої співпраці, вони вчаться застосовувати знання на практиці, працювати в команді над конкретним завданням, презентувати свої результати;
* методи змішаного навчання, які поєднують у собі традиційне й дистанційне навчання та найбільше відповідають інтересам і вподобанням учнів, які живуть у період стрімкого інформаційно-технологічного розвитку суспільства;
* звернення до досвіду учнів, що гарантує перетворення кожного учня на справжнього учасника освітнього процесу, співтворця й конструктора нових знань;
* відповідні форми оцінювання, а саме: самооцінювання, яке формує здатність до самоаналізу, спостережливість за собою, вміння бачити та визнавати власні помилки; взаємооцінювання, що виховує відкритість до критики з боку інших, здатність відокремлювати об’єкт від суб’єкта оцінювання, вміння слухати, аналізувати й порівнювати.

Планування та організація навчальної діяльності проводиться на основі базових цінностей, загальних компетенцій з урахуванням цілей, змісту та очікуваних результатів навчання, які зазначені в навчальних програмах, підтримки інтеграції з іншими предметами та суб’єктами навчальної діяльності, змістовими лініями курсу інформатики. Очікувані результати навчання вчитель визначає відповідно до складових компетентностей, зазначених у програмі з інформатики.

Програма не обмежує самостійність та творчу ініціативу вчителя, передбачаючи гнучкість вибору та розподілу навчального матеріалу відповідно до потреб учнів та обраних засобів навчання. Вона не встановлює кількість годин та порядок вивчення тем у рамках навчального року, а лише вказує на очікувані результати навчання та зміст навчального матеріалу, вивчення якого є об’єктом тематичного оцінювання. Учитель може розподіляти навчальний час на власний розсуд, ураховуючи особливості наявного матеріально-технічного забезпечення, попередній досвід, рівень знань учнів та інші фактори. За необхідності педагог може змінювати порядок вивчення тем, не порушуючи при цьому змістових та логічних зв’язків між ними та враховуючи, що на вивчення змістової лінії «Алгоритми та програми» має приділятися не менше 40% загального навчального часу у 5–8 класах і не менше 30% навчального часу у 9 класі.

Звертаємо увагу, що хоча з програми вилучені розділи узагальнення та повторення матеріалу, а також резервні години, учитель може передбачити необхідний на його думку час для повторення як на початку, так і наприкінці навчального року або півріччя.

**Усі уроки курсу інформатики передбачають практичну роботу учнів за комп’ютером.** У практичних завданнях слід передбачати використання актуального для учнів змістового матеріалу й завдань з інших предметних галузей.

Проектну діяльність та розв’язування компетентнісних задач у програмі інформатики можна застосовувати під час вивчення різних тем. Виконання навчальних проектів дозволяє вчителю розширити рамки теми, а учневі – проявити свої творчі здібності. Проектні завдання в курсі інформатики виконуються в невеликих групах, а компетентнісні – індивідуально. Таким способом учням надається можливість практичного використання отриманих у межах теми (курсу) вмінь. Результати означеної діяльності необхідно представити у вигляді закінченого інформаційного продукту для того, щоб учні могли порівнювати свої роботи і навчатися один в одного у процесі публічної презентації виконаних робіт перед класом. Під час представлення проекту оцінюється:

* планування дослідження, творчість і раціональність запропонованого розв’язання;
* досягнення результатів навчання та компетенцій;
* технічні характеристики, естетика й оригінальність розроблених матеріалів;
* розвиток та самонавчання учня.

Компетентнісно орієнтовані завдання у своєму змісті містять:

* мотивацію (стимул), що є введенням у проблему (практично-орієнтовану) і відповідає на запитання «з якою метою треба це робити?»;
* формулювання завдання – відповідає на запитання «що саме треба зробити?». Учень для себе має чітко визначити суть завдання: відповісти на запитання, систематизувати початкові дані, підібрати необхідні інформаційні ресурси та програмні засоби, оцінити доцільність їх використання тощо;
* інформацію (додаткову), необхідну для розв’язання задачі. Ця частина відповідає на запитання «чому?»;
* перевірку (критерії) – результат виконання, відповідає на запитання «що, в якій формі потрібно зазначити?».

У процесі вивчення інформатики у старшій школі важливим фактором є самостійна навчальна діяльність учнів у способі навчання, у способі перенесення учнями результатів навчання на більш широкий контекст. У зв’язку з цим необхідно надавати учням можливість учитися самостійно та разом з іншими учасниками освітнього процесу (індивідуальні, парні та групові роботи) для підтримки їх активності. Під час самостійної навчальної діяльності формується особиста відповідальність учня за вибір засобів інформаційно-комунікаційних технологій для досягнення навчальних цілей. Вони дають змогу індивідуалізувати процес навчання та об’єднати різноманітні види роботи – групову, самостійну, дистанційну.

**Перелік деяких корисних ресурсів для самоосвіти учнів**

Електронні підручники, курси:

<http://www.ed-era.com>

<http://disted.edu.vn.ua/>

<http://itknyga.com.ua/index/bezkoshtovno/0-19>

<https://dystosvita.gnomio.com/>

Ресурси для навчання програмуванню:

<https://blockly-games.appspot.com/>

<https://code.org/>

<https://www.e-olymp.com/uk/>

http://scratch.mit.edu/[projects](http://scratch.mit.edu/projects/editor)/editor

Для практичних робіт при вивченні відповідних тем та для розвитку в учнів навичок алгоритмічного мислення радимо використовувати інтерактивні та ігрові задачі міжнародного конкурсу з інформатики «Бобер» попередніх років: <http://bober.net.ua/page.php?name=archive&>, завдання якого можна застосовувати під час вивчення всього курсу інформатики – із 2-го по 11-й класи.

Також корисним є безкоштовний інформаційний ресурс <http://thefuture.tilda.ws/about>, на якому розміщуються науково-популярні статті про новітні технології для учнів та вчителів. Напрями, що охоплюють матеріали ресурсу: 3D-друк, Інтернет речей, Розумний дім, використання дронів та інше.

Для вчителів інформатики рекомендується використовувати безкоштовний масовий відкритий онлайн-курс «[Алгоритми і проекти Scratch](https://edx.prometheus.org.ua/courses/course-v1%3AKPI%2BScratch101%2B2017_T1/about)» на українській платформі масових відкритих онлайн-курсів “Prometheus” ([https://edx.prometheus.org.ua/courses/course-v1:KPI+Scratch101+2017\_T1/about](https://edx.prometheus.org.ua/courses/course-v1%3AKPI%2BScratch101%2B2017_T1/about)).

***Оцінювання навчальних досягнень учнів***

Варто зауважити, що впровадження компетентнісного підходу зумовлює переосмислення технологій контролю й оцінювання: із оцінювання предметних знань, умінь і навичок до оцінювання компетентностей, зокрема готовності та здатності учнів застосовувати здобуті знання і сформовані навички у своїй практичній діяльності. Тепер о*б’єктом оцінювання* навчальних досягнень учнів з інформатики є рівень розвитку їх компетентностей, які інтегрують знання, вміння, навички, досвід творчої діяльності та емоційно-ціннісне ставлення до навколишньої дійсності. При оцінюванні навчально-пізнавальної діяльності учнів варто збалансовано оцінювати всі три компоненти, що відповідають складникам компетентності: діяльнісний (діяльність/уміння), знаннєвий (знання), ціннісний (ставлення). Навчальна програма розрахована на те, що при вивченні кожної теми формуються як технологічні навички/вміння, так і ціннісне ставлення до сучасних інформаційних технологій та їх вплив на суспільство й особистість. Знаннєвий складник включає перелік обов’язкових термінів і понять, якими учень оперуватиме після вивчення кожної теми. **Протягом вивчення теми мають проводитися оцінювані практичні роботи, результати яких повинні враховуватися під час виставлення тематичної оцінки.**

Форми оцінювання при вивченні інформатики:

* виконання завдань практичного змісту;
* тестування за допомогою програмних засобів або онлайнових сервісів;
* урахування особистих досягнень в опануванні інформаційних технологій;
* співбесіда (інтерв’ю) як доповнення до тестування або практичної роботи;
* взаємоконтроль учнів у парах або групах та самооцінка.

***Організація діяльності на уроках інформатики***

Умови навчання повинні забезпечувати ефективне засвоєння учнями програмового матеріалу та відповідати вимогам щодо безпеки життєдіяльності учасників освітнього процесу, що наведені в Державних санітарних правилах і нормах влаштування, утримання загальноосвітніх навчальних закладів та організації навчально-виховного процесу ДСанПіН 5.5.2.008-01, Правилах пожежної безпеки для навчальних закладів та установ системи освіти України та Правилах безпеки під час навчання в кабінетах інформатики навчальних закладів системи загальної середньої освіти. Обладнання навчального приміщення (класу, кабінету) має відповідати вимогам (технічним, педагогічним тощо) Положення про кабінет інформатики та інформаційно-комунікаційних технологій навчання загальноосвітніх навчальних закладів, Типового переліку комп’ютерного обладнання для закладів дошкільної, середньої та професійної освіти (наказ МОН від 02.11.2017 № 1440).

Відповідно до листа МОН від 17.07.2013 № 1/9-497 «Про використання Інструктивно-методичних матеріалів із питань створення безпечних умов для роботи в кабінетах інформатики та інформаційно-комунікаційних технологій загальноосвітніх навчальних закладів» щороку перед початком роботи учнів у кабінеті інформатики вчитель проводить первинний інструктаж із безпеки життєдіяльності, який ознайомлює учнів із правилами поведінки в кабінеті.

Звертаємо увагу, що Державні санітарні правила та норми «Влаштування і обладнання кабінетів комп’ютерної техніки в навчальних закладах та режим праці учнів на персональних комп’ютерах ДСанПіН 5.5.6.009-98» втратили чинність на підставі наказу МОЗ від 26.06.2017 № 709. Насамперед це свідчить про те, що на даний момент ***тривалість безперервної роботи за комп’ютером учнів нормативно не регламентується.***

**Про те, це не означає, що вчитель не повинен дотримуватися методичної структури проведення будь-якого уроку.**