

Рішенням Колегії МОН України від 27.06.2013 року
протокол №2/4-2 програмі надано гриф
«Затвержено Міністерством освіти і науки»

ПРОГРАМА КУРСУ «ІНФОРМАТИКА» для 8–9 класів загальноосвітніх навчальних закладів з поглибленим вивченням інформатики

Вступ

Ця програма розроблена відповідно до Державного стандарту базової і повної загальної середньої освіти, затвердженого постановою Кабінету Міністрів України від 23 листопада 2011 року №1392 і спрямована на реалізацію мети і завдань освітньої галузі «Технології», визначених у ньому.

Програма призначена для переважного використання у спеціалізованих школах із поглибленим вивченням предметів технічного (інженерного) циклу і спеціалізованих школах, гімназіях, ліцеях, колегіумах, класах з поглибленим вивченням окремих предметів, зокрема природничо-математичного і технічного (інженерного) спрямування (Додаток 5 і Додаток 8 до Наказу МОНмолодьспорт №409 від 03.04.12 року «Про затвердження Типових навчальних планів загальноосвітніх навчальних закладів II ступеня»).

Навчальна програма складається з: пояснювальної записки, у якій визначено мету й завдання навчання інформатики в основній школі, схарактеризовано надпредметну й предметну складові ІКТ-компетентностей, структуру навчального процесу й умови навчання інформатики, подано розподіл навчальних годин на вивчення розділів програми; зміст навчального матеріалу, передбачувані результати навчання й критерії оцінювання навчальних досягнень учнів.

Пояснювальна записка

Інформатика — фундаментальна наука про методи, засоби й технології опрацювання інформації, яка разом з речовиною й енергією належить до основних понять, на яких будується сучасна наукова картина світу. Інформатика є теоретичною платформою розроблення й використання інформаційних і комунікаційних технологій, які є одним із найголовніших досягнень сучасної цивілізації і важливою продуктивною силою сучасного суспільства.

Мета і завдання навчального курсу

Мета навчання інформатики в основній школі полягає в розвитку особистості учнів, підготовці їх до активного життя й праці в інформаційному суспільстві.

Мета досягається завдяки формуванню в учнів наукового світогляду, інформаційної культури, алгоритмічного й критичного стилів мислення, інформатичних і ключових компетентностей, розвитку творчих здібностей, умінь і навичок працювати із сучасними засобами інформаційних і комунікаційних технологій.

Зміст навчального предмету спрямовано на опанування учнями наукових основ інформатики, відповідних фундаментальних понять, принципів побудови й функціонування засобів інформаційних і комунікаційних технологій, оволодіння ними.

Завдання навчання предмету інформатика в основній школі полягають у тому, щоб:

- сформувати в учнів базові знання про інформацію та інформаційні процеси, значення інформації і знань на сучасному етапі розвитку інформаційного суспільства;
- надати учням основні історичні відомості про розвиток інформатики і засобів обчислювальної техніки, внесок зарубіжних і вітчизняних учених у їх розвиток;
- сформувати в учнів уміння використовувати програмні й інформаційно-комунікаційні засоби в навчальній та повсякденній діяльності;
- забезпечити оволодіння учнями основами інформаційної культури;
- сформувати вміння творчо виконувати навчальні завдання, розробляти раціональні алгоритми виконання і здійснювати аналіз їх виконання;
- сформувати в учнів початкові навички програмування, уміння налагоджувати програми й аналізувати отримані результати.

Виконання зазначених завдань забезпечує формування в учнів наукового світогляду, інформаційної культури, алгоритмічного й критичного стилів мислення, розвитку творчих здібностей, умінь і навичок працювати із сучасними засобами інформаційних і комунікаційних технологій.

Набуті учнями в процесі навчання інформатики знання, уміння і навички забезпечать підґрунтя для формування в учнів предметної *інформатичної компетентності* і *ключових компетентностей*, передбачених Державним стандартом базової і повної загальної середньої освіти.

Набуті учнями в результаті навчання компетентності надають їм змогу:

- пояснювати властивості інформації й закономірності інформаційних процесів;
- оволодіти основними методами наукового пізнання;
- розуміти наукові основи опрацювання відомостей, застосовувати основні поняття, пов'язані з алгоритмізацією опрацювання даних, управління об'єктами і процесами;
- бути готовим до активної життєдіяльності в умовах інформаційного суспільства, стати в майбут-

ньому не лише повноцінним членом такого суспільства, а й його творцем;

- використовувати набуті знання в подальшій навчальній та практичній діяльності в умовах інформаційного суспільства.

Структура навчальної програми

Курс поглибленого вивчення предмету «Інформатика» розрахований на 280 годин за рахунок інваріантної і варіативної частин навчального плану (табл. 1).

Решту годин варіативної частини навчального плану рекомендується використовувати для навчання (у формі факультативів) розділів математики, які необхідні для кращого сприйняття учнями навчального матеріалу з інформатики.

Відповідно до Державного стандарту базової і повної загальної середньої освіти курс «Інформатика» будуватиметься за такими *змістовими лініями*:

- інформація, інформаційні процеси, системи, технології;
- комп'ютер як універсальний пристрій для опрацювання даних;
- комп'ютерні мережі;
- інформаційні технології створення та опрацювання текстових документів, графічних зображень, числових даних, об'єктів мультимедіа, мультимедійних презентацій, систем управління базами даних;
- комп'ютерне моделювання;
- основи алгоритмізації та програмування.

У програмі конкретизовано *зміст навчального матеріалу* для кожного класу і подано відповідні *вимоги до навчальних досягнень* учнів. Перелік вимог зорієнтує вчителя на досягнення мети навчання за кожною темою програми, полегшить планування мети і завдань навчання на уроках, надасть змогу виробити адекватні методичні підходи до проведення навчальних занять, поточного й тематичного оцінювання.

Зміст навчання інформатики структуровано за темами з визначенням кількості годин на їх вивчення. Такий розподіл змісту і навчального часу є орієнтовним. Учителю та авторам підручників надається право коригувати послідовність вивчення тем залежно від методичної концепції та конкретних навчальних ситуацій, від рівня підготовки учнів і сформованості у них предметної ІКТ-компетентності, вибудовуючи найбільш доречну для конкретного навчального закладу або класу траєкторію навчання.

Запропонована кількість часу на вивчення кожної теми також є орієнтовною, вчитель може її змінювати (до 15% від загального навчального часу для кожного класу). При цьому вчитель має забезпечити рівень навчальних досягнень учнів, зазначених у програмі з кожної теми.

Характеристика умов навчання. Успішне впровадження навчального курсу «Інформатика» залежить від обов'язкової саме для нього складової — *стандарту можливостей для навчання*, в якому зазначаються обов'язкові *умови та ресурси*, потрібні для реалізації державних вимог до рівня загальноосвітньої підготовки учнів середнього шкільного віку.

До *обов'язкових умов* успішного впровадження курсу слід віднести:

- підготовленість учителів інформатики до поглибленого викладання курсу «Інформатика»;
- забезпечення навчального закладу сучасною комп'ютерною технікою й програмним забезпеченням відповідно до чинних норм (специфікацій навчальних комп'ютерних комплексів (НКК));
- під'єднання до Інтернету всіх комп'ютерів НКК, якість якого забезпечує виконання кожним учнем завдань роботи із сервісами Інтернету, та наявність у навчальному закладі локальної комп'ютерної мережі.

Під час вивчення предмета кожний урок проводиться з використанням комп'ютерів, тому на кожному уроці класи діляться на підгрупи так, щоб кожен учень був забезпечений індивідуальним робочим місцем за комп'ютером, але не менш як 8 учнів у підгрупі.

Перелік необхідних *програмних засобів*:

- операційна система з графічним інтерфейсом;
- програма для запису даних на оптичні носії;
- архіватор;
- антивірусна програма;
- векторний графічний редактор;
- растровий графічний редактор;
- текстовий процесор;
- редактор презентацій;
- редактор публікацій;
- програми для опрацювання об'єктів мультимедіа;
- табличний процесор;
- система управління базами даних;
- програма для створення карт знань;
- програми для опрацювання аудіо- та відеоданих і розроблення потокових презентацій;
- електронні словники та програми-перекладачі;
- система розпізнавання текстів;
- веб-браузер;
- навчальне середовище виконання алгоритмів;
- програми для розвитку логічного та критичного мислення;
- розвивальні програми;
- комп'ютерні програми для підтримки вивчення різних навчальних предметів;
- системи програмування.

Вибір певних операційних систем, програмних та апаратних платформ, програмних засобів здійснює вчитель.

Таблиця 1

Клас	Кількість годин на тиждень		Загальна кількість годин	
	Інваріантна частина	Варіативна частина	Інваріантна частина	Варіативна частина
8 клас	2	2	70	70
9 клас	2	2	70	70
Всього	280			

Розподіл навчальних годин на вивчення розділів програми

Таблиця 2

№	Назва розділу	Класи і кількість годин		Всього
		8 кл.	9 кл.	
1	Інформаційні технології у суспільстві	0,0	3,0	3,0
2	Математичні основи обчислювальної техніки	6,0	0,0	6,0
3	Кодування даних	10,0	0,0	10,0
4	Комп'ютер як універсальний пристрій для опрацювання даних	12,0	0,0	12,0
	Інформаційні технології			
5.1	Створення та опрацювання текстових документів (електронний документообіг, комп'ютерні публікації)	8,0	12,0	20,0
5.2	Створення та опрацювання графічних зображень	6,0	0,0	6,0
5.3	Створення та опрацювання об'єктів мультимедіа	8,0	0,0	8,0
5.4	Створення та опрацювання мультимедійних презентацій	8,0	0,0	8,0
5.5	Технології опрацювання числових даних у середовищі табличного процесора	14,0	0,0	14,0
5.6	Система управління базами даних	0,0	16,0	16,0
7	Мережні технології	0,0	18,0	18,0
8	Основи інформаційної безпеки	0,0	4,0	4,0
9	Створення персонального навчального середовища	0,0	6,0	6,0
10	Моделювання	0,0	8,0	8,0
11	Основи алгоритмізації та програмування	56,0	56,0	112,0
12	Розв'язування компетентнісних задач, виконання індивідуальних і групових навчальних проєктів	8,0	12,0	20,0
13	Резерв часу	4,0	5,0	9,0
	Всього	140	140	280

Зміст навчального матеріалу
8 клас, 140 год. (4 год. на тиждень)

Таблиця 3

Зміст навчального матеріалу	Передбачувані результати навчання
<p>1. Математичні основи обчислювальної техніки (6 год.) Поняття про системи числення. Позиційні і непоозиційні системи числення. Переведення чисел з десятичної системи числення в довільну і навпаки. Двійкова та шістнадцяткова системи числення. Арифметичні операції в двійковій і шістнадцятковій системах числення. Перетворення чисел з двійкової системи числення в шістнадцяткову і навпаки. Практична робота №1. Переведення чисел з десятичної системи числення в іншу і навпаки, з двійкової в шістнадцяткову і навпаки. Операції над числами в двійковій і шістнадцятковій системах числення</p>	<p>Учень (учениця) пояснює: • поняття системи числення; пояснює зміст понять: • позиційна і непоозиційна система числення; • основа позиційної системи числення; наводить приклади: • запису чисел в різних системах числення; описує: • позиційні та непоозиційні системи числення; формулює: • правила переведення з десятичної системи числення довільну і навпаки; • правила переведення з двійкової системи числення шістнадцяткову і навпаки; розрізняє: • позиційні і непоозиційні системи числення; вміє: • записувати числа в позиційній системі числення з довільною основою; • переводити числа з десятичної системи числення довільну і навпаки; • виконувати операції додавання, віднімання множення над натуральними числами у двійковій шістнадцятковій системі числення; • переводити числа з двійкової системи числення в шістнадцяткову і навпаки</p>
<p>2. Кодування даних (10 год.) Опрацювання даних як інформаційний процес. Кодування та декодування повідомлень. Одиниці вимірювання довжини двійкового коду. Кодування символів. Кодування графічних даних. Поняття колірної схеми. Кодування звукових даних. Кодування відео даних. Практична робота №2. Розв'язування задач на визначення довжини двійкового коду даних різних типів</p>	<p>використовує: • одиниці вимірювання довжини двійкового коду; пояснює зміст понять: • кодування повідомлень, двійкове кодування; • колірна схема; наводить приклади: • колірних схем; описує: • принципи кодування символів, графічних і звукових даних; відео даних; • загальну структуру таблиць кодів, наприклад, ASCII, Windows — 1251, Unicode; • принципи кодування кольору в колірних схемах RGB, CMYK, HSB; порівнює: • таблиці кодів символів; колірні схеми; розрізняє: • одиниці вимірювання довжини двійкового коду: біт, байт, кілобайт, мегабайт, гігабайт, терабайт; • числа у різних системах числення; розв'язує: • задачі на кодування повідомлень; вимірювання довжини двійкового коду повідомлень;</p>

	<p>вміє:</p> <ul style="list-style-type: none"> • кодувати і декодувати повідомлення за певними правилами; • визначати довжину двійкового коду даних різних типів; • переходити при визначенні довжини коду повідомлення від одних одиниць вимірювання до інших
<p>3. Комп'ютер як універсальний пристрій для опрацювання даних (12 год.)</p> <p>Архітектура комп'ютера. Процесор, його будова та призначення. Пам'ять комп'ютера, її види. Зовнішні та внутрішні запам'ятовуючі пристрої.</p> <p>Будова і алгоритм роботи ЕОМ за фон Нейманом.</p> <p>Пристрої введення / виведення даних.</p> <p>Пристрої, що входять до складу мультимедійного обладнання. Технічні характеристики складових комп'ютера.</p> <p>Визначення значень властивостей комп'ютера.</p> <p>Історія засобів опрацювання інформаційних об'єктів. Покоління електронних обчислювальних машин (ЕОМ).</p> <p>Класифікація та загальна характеристика програмного забезпечення. Ліцензії на програмне забезпечення, їх типи.</p> <p>Проблеми сумісності програмного забезпечення.</p> <p>Класифікація, основні функції та складові операційних систем.</p> <p>Поняття про ядро операційної системи, інтерфейс користувача, драйвери та утиліти.</p> <p>Інсталяція програмного забезпечення.</p> <p>Інсталяція та деінсталяція компонентів операційної системи.</p> <p>Архівування даних. Стиснення даних, види стиснення даних. Архіватори. Типи архівів. Операції над архівами.</p> <p>Запис даних на оптичні носії. Форматування та копіювання дисків.</p> <p>Дефрагментація пристроїв пам'яті з файловими системами, встановлення розкладу її проведення.</p> <p>Практична робота №3. Конфігурування комп'ютера під потребу.</p> <p>Практична робота №4. Створення архівів та операції над ними</p>	<p>пояснює:</p> <ul style="list-style-type: none"> • призначення складових комп'ютера; • класифікацію та загальні характеристики програмного забезпечення; • види ліцензій на програмне забезпечення; <p>має уявлення:</p> <ul style="list-style-type: none"> • про методи стиснення даних; <p>пояснює:</p> <ul style="list-style-type: none"> • відмінність між оперативним і постійним запам'ятовувачами пристроями; • відмінність між зовнішніми та внутрішніми запам'ятовувачами пристроями; • призначення пристроїв, що входять до складу мультимедійного обладнання; • сутність форматування носіїв даних; <p>описує:</p> <ul style="list-style-type: none"> • типову архітектуру персонального комп'ютера; • основні властивості запам'ятовувачів пристроїв; • функціональне призначення, основні складові частини та основні характеристики властивості процесорів; • призначення та основні властивості моніторів і відеоадаптерів; • основні властивості принтерів; • принцип дії накопичувачів на магнітних та оптичних дисках; • історію виникнення засобів опрацювання інформаційних об'єктів; • основні властивості і можливості ЕОМ різних поколінь; • види сучасних комп'ютерів і їх застосування; • призначення програмного забезпечення різних типів; <p>наводить приклади:</p> <ul style="list-style-type: none"> • пристроїв введення, виведення, зберігання та опрацювання даних; • технічних значень властивостей складових комп'ютера; <p>розрізняє:</p> <ul style="list-style-type: none"> • файли архівів різних типів; • різні види ліцензій на програмне забезпечення; <p>класифікує:</p> <ul style="list-style-type: none"> • процесори; • запам'ятовуючі пристрої; • пристрої введення та виведення даних; • сучасні комп'ютери; • програмне забезпечення; <p>уміє:</p> <ul style="list-style-type: none"> • визначати значення властивостей комп'ютера; • конфігурувати комп'ютер під потребу; • інсталювати та деінсталювати програмне забезпечення; • інсталювати та деінсталювати програмне складові операційної системи; • створювати архіви різних типів; • додавати дані до архівів, знаходити дані в архівах, вилучати дані з архівів; оновлювати архіви; • записувати дані на оптичні диски; • формувати зовнішні носії даних; • створювати копії оптичних дисків; • виконувати дефрагментацію носіїв даних
<p>4. Створення та опрацювання текстових документів (8 год.)</p> <p>Формати файлів текстових документів.</p> <p>Створення та форматування списків, таблиць, колонок у текстовому документі.</p> <p>Стильове оформлення абзаців.</p> <p>Створення нових стилів і їх використання в інших документах. Рівні абзаців та їх зв'язок зі стилями.</p> <p>Шаблони документів.</p> <p>Розділи. Структура документа. Колонтитули.</p>	<p>Учень розрізняє</p> <ul style="list-style-type: none"> • формати файлів текстових документів; <p>описує поняття:</p> <ul style="list-style-type: none"> • шаблон текстового документа; • структура документа; <p>пояснює:</p> <ul style="list-style-type: none"> • стильове оформлення документів різних типів та їх об'єктів; • призначення колонтитулу; • призначення покажчика; <p>класифікує:</p> <ul style="list-style-type: none"> • списки; <p>вміє створювати в текстовому документі:</p> <ul style="list-style-type: none"> • нумеровані й марковані списки;

<p>Посилання. Закладки та виноски. Автоматизоване створення змісту та покажчиків. Алгоритм опрацювання складного текстового документа. Макроси та їх використання. Робота з кількома документами. Налаштування параметрів роботи середовища текстового процесора. Практична робота №5. Створення текстового документа, що містить об'єкти різних типів. Практична робота №6. Використання стилів для оформлення текстових документів. Практична робота №7. Структура документа. Автоматизоване створення змісту та покажчиків. Макроси</p>	<ul style="list-style-type: none"> ● таблиці; ● колонки; ● колонтитули та налаштовувати параметри сторінок; ● зміст документа в автоматичному режимі; ● покажчики; <p>використовує:</p> <ul style="list-style-type: none"> ● стилі символів та абзаців для форматування тексту; ● шаблони документів; ● інструменти для креслення й налаштування властивостей таблиць у текстовому документі; ● посилання в текстовому документі; ● структуру документа для перегляду його змісту та навігації в ньому; ● макроси;
<p>5. Комп'ютерна графіка. Векторний графічний редактор (6 год.) Поняття комп'ютерної графіки. Растрові зображення, їх властивості. Формати файлів растрових зображень. Векторні зображення, їх властивості. Формати файлів векторних зображень. Векторний графічний редактор. Особливості побудови й опрацювання векторних зображень. Засоби векторного графічного редактора. Алгоритм побудови зображення з графічних примітивів. Створення малюнків з кривих і ламаних. Додавання тексту до графічних зображень та його форматування. Групкування і вирівнювання об'єктів. Багатшарові зображення, розміщення об'єктів у шарах. Практична робота №8. Створення простих векторних зображень. Практична робота №9. Створення складених векторних зображень</p>	<p>пояснює зміст понять:</p> <ul style="list-style-type: none"> ● комп'ютерна графіка; ● растрове зображення; ● векторне зображення; <p>розрізняє:</p> <ul style="list-style-type: none"> ● призначення векторного графічного редактора; ● формати файлів векторної та растрової графіки; ● особливості побудови векторних зображень; ● способи створення зображень із графічних примітивів; <p>порівнює:</p> <ul style="list-style-type: none"> ● властивості векторних і растрових зображень; <p>уміє:</p> <ul style="list-style-type: none"> ● створювати й редагувати векторні зображення; ● створювати об'єкти, що складаються з базових графічних примітивів; ● обертати, відображати й масштабувати об'єкти; ● зафарбовувати об'єкти, використовуючи однорідні, градієнтні, візерункові й текстурні заливки; ● додавати до графічних зображень текст та формувати його; ● групувати і вирівнювати об'єкти; ● використовувати шари для створення зображень; <p>використовує:</p> <ul style="list-style-type: none"> ● інструменти малювання, призначені для креслення ліній, стрілок, основних геометричних фігур; ● лінійки, сітку
<p>6. Створення та опрацювання об'єктів мультимедіа (8 год.) Формати аудіо- та відеофайлів. Конвертація аудіо та відеофайлів. Програмне забезпечення для опрацювання об'єктів мультимедіа. Захоплення аудіо та відео, створення аудіо-, відео фрагментів з використанням цифрових відео або фотокамер. Побудова аудіо- та відеоряду. Додавання до відеокліпу відеоефектів та налаштування переходів між його фрагментами. Налаштування часових параметрів аудіо- та відеоряду. Збереження створених відеофільмів на носіях даних. Сервіси публікування відеофайлів. Подкастинг. Практична робота №10. Захоплення та конвертування аудіо (відео) даних. Практична робота №11. Створення відеокліпу. Додавання відеоефектів, налаштування часових параметрів аудіо- та відеоряду</p>	<p>розрізняє:</p> <ul style="list-style-type: none"> ● формати аудіо- та відеофайлів; <p>пояснює:</p> <ul style="list-style-type: none"> ● призначення програмного забезпечення опрацювання аудіо- та відеофайлів; ● призначення подкастів; ● призначення сервісів публікування відео файлів; <p>порівнює:</p> <ul style="list-style-type: none"> ● формати аудіо- та відеофайлів; ● режими відображення відеоряду; <p>наводить приклади:</p> <ul style="list-style-type: none"> ● програмного забезпечення опрацювання аудіо- та відеофайлів; ● сервісів публікування відеофайлів та створення подкастів; <p>вміє:</p> <ul style="list-style-type: none"> ● розробляти сценарій відеокліпу; ● імпортувати у відеокліп аудіо- та відеофрагменти із зовнішніх джерел; ● синхронізувати відеоряд з аудіорядом; ● налаштовувати часові параметри аудіо- та відеоряду; ● додавати відеоефекти до відеокліпу та налаштовувати переходи між його фрагментами; ● публікувати відеофайли в Інтернеті.

<p>7. Мультимедійні презентації (8 год.) Етапи розробки презентації. Критерії оцінювання презентації. Макети слайдів. Стильове оформлення слайдів презентації. Елементи дизайну презентацій. Використання організаційних діаграм у презентаціях. Проектування та розробка розгалужених презентацій. Гіперпосилання і елементи керування в презентаціях. Додавання відеокліпів, звукових ефектів і мовного супроводу до слайдової презентації. Елементи анімації. Вбудовані та зв'язані об'єкти в презентаціях. Керування показом презентації. Друкування презентації. Практична робота №12. Проектування та розробка розгалужених презентацій за визначеними критеріями. Використання вбудованих та зв'язаних об'єктів у презентації. Практична робота №13. Розробка презентацій з елементами анімації, відеокліпами, звуковими ефектами та мовним супроводом</p>	<p>пояснює зміст понять:</p> <ul style="list-style-type: none"> ● макет слайда; ● дизайн слайда та презентації; ● структура презентації; <p>пояснює:</p> <ul style="list-style-type: none"> ● критерії оцінювання презентації; ● вибір стильового оформлення слайдів презентації; ● призначення елементів управління в презентаціях; ● особливості використання вбудованих і зв'язаних об'єктів; <p>розрізняє:</p> <ul style="list-style-type: none"> ● вбудовані і зв'язані об'єкти; описує: ● спосіб застосування ефектів анімації до процесу змінення слайдів та об'єктів на слайдах; ● способи показу презентацій; ● алгоритм вбудовування та зв'язування об'єктів; <p>вміє:</p> <ul style="list-style-type: none"> ● проектувати і розробляти структуру презентації; ● добирати стильове оформлення презентації та дизайн слайдів; ● додавати до слайдів організаційні діаграми; ● додавати до слайдів ідео кліпи, звукові ефекти, мовний супровід; ● додавати до об'єктів на слайдах анімаційні ефекти; ● налаштовувати анімаційні ефекти змінення слайдів; ● використовувати гіперпосилання і елементи управління в процесі створення і демонстрації презентації; ● вбудовувати об'єкти; ● підготовлювати до друку і друкувати слайди презентації; ● оцінювати презентацію за заданими критеріями
<p>8. Технології опрацювання числових даних у середовищі табличного процесора (14 год.) Типи величин у текстовому процесорі. Уведення даних. Автозаповнення. Абсолютні, відносні й мішані посилання на комірки і діапазони комірок. Копіювання формул і модифікація посилань під час копіювання. Обчислювальні алгоритми в середовищі табличного процесора. Призначення й використання основних математичних, статистичних, логічних функцій табличного процесора. Умове форматування. Створення та налагодження діаграм різного типу, вибір типу діаграми. Упорядкування даних у таблицях. Автоматичні та розширені фільтри. Проміжні підсумки. Створення форм уведення-виведення. Практична робота №14. Розв'язування задач на обчислення. Обчислювальні алгоритми в середовищі табличного процесора. Практична робота №15. Використання математичних, логічних та статистичних функцій табличного процесора. Умове форматування. Практична робота №16. Упорядкування даних у таблицях. Автоматичні та розширені фільтри. Практична робота №17. Створення форм уведення-виведення</p>	<p>наводить:</p> <ul style="list-style-type: none"> ● алгоритми виконання обчислень у середовищі табличного процесора; <p>описує:</p> <ul style="list-style-type: none"> ● критерії визначення типу діаграми для відображення одного або кількох рядів даних; ● мету та способи фільтрування даних у таблицях; ● обчислювальні алгоритми в середовищі табличного процесора; <p>пояснює:</p> <ul style="list-style-type: none"> ● використання абсолютних, відносних і мішаних посилань на комірки та діапазони комірок; ● використання формул і використання посилань на комірки і діапазони у формулах; ● призначення та алгоритм застосування умовного форматування; ● призначення та відмінності автофільтра та розширеного фільтра; ● алгоритм створення проміжних підсумків; <p>розрізняє:</p> <ul style="list-style-type: none"> ● автоматичні та розширені фільтри; ● типи діаграм відповідно до мети їх застосування; <p>інтерпретує:</p> <ul style="list-style-type: none"> ● дані та відомості, що відображаються у процесі побудови діаграм різного типу; <p>наводить приклади:</p> <ul style="list-style-type: none"> ● математичних, статистичних, логічних функцій табличного процесора; ● використання умовного форматування в процесі розв'язування задач; <p>уміє:</p> <ul style="list-style-type: none"> ● записувати абсолютні, відносні та мішані посилання в комірках і діапазони комірок; ● застосовувати основні математичні, статистичні, логічні функції для аналізу й опрацювання даних; ● визначати тип діаграми для відображення одного чи кількох рядів даних; ● змінювати тип і формувати діаграми; ● задавати діапазон вхідних даних для діаграми й діапазон даних для кожного ряду; ● налаштовувати параметри відображення діаграми, поля даних та рядів даних; ● упорядковувати дані в таблицях за значеннями одного чи кількох полів; ● виконувати умове форматування; ● використовувати автоматичні і розширені фільтри; ● знаходити проміжні підсумки; ● використовувати шаблони електронних таблиць;

	<ul style="list-style-type: none"> • описувати складені критерії пошуку для застосування розширеного фільтру; • використовувати автозаповнення клітинок для прискорення введення даних;
Основи алгоритмізації та програмування, 56 годин	
<p>9. Основні поняття алгоритмізації (4 год.) Алгоритм та основні поняття алгоритмізації; властивості алгоритмів; виконавець та система команд виконавця; формальне виконання алгоритму; способи описання алгоритмів; базові алгоритмічні структури; типи алгоритмів; аргументи, результати, проміжні величини.</p> <p>Практична робота №18. Способи представлення алгоритмів. Базові алгоритмічні структури. Типи алгоритмів</p>	<p>називає: основні поняття алгоритмізації; властивості алгоритмів; способи представлення алгоритмів; базові алгоритмічні структури; типи алгоритмів; аргументи, результати, проміжні величини;</p> <p>наводить: власні приклади алгоритмів та виконавців алгоритмів; аргументів, результатів, проміжних величин;</p> <p>розпізнає: відмінність між різними базовими алгоритмічними структурами та типами алгоритмів; аргументами, результатами, проміжними величинами;</p> <p>характеризує: задані алгоритми за типами; способами представлення;</p> <p>описує: задані алгоритми різними способами;</p> <p>пояснює: сутність поняття «виконавець алгоритму» та «система команд виконавця»; сутність формального виконання алгоритму; поняття аргументу, результату та проміжної величини;</p> <p>формулює: визначення алгоритму, базових алгоритмічних структур; характерні ознаки різних типів алгоритмів, аргументів, результатів та проміжних величин;</p> <p>обґрунтовує: наявність тих чи інших базових алгоритмічних структур у запропонованому алгоритмі; належність конкретного алгоритму до відповідного типу; коректність дій при виконанні запропонованого алгоритму; необхідність використання у запропонованому алгоритмі аргументів, результатів та проміжних величин;</p> <p>порівнює: різні способи представлення алгоритмів; різні алгоритми за типами; змінні як аргументи, результати та проміжні величини;</p> <p>класифікує: алгоритми за способами представлення та за типами; змінні як аргументи, результати та проміжні величини;</p> <p>аналізує: належність запропонованих алгоритмів до тих чи інших типів; визначення змінних як аргументів, результатів або проміжних величин;</p> <p>висловлює судження: щодо наявності у запропонованій послідовності дій властивостей алгоритму; наявності у запропонованих алгоритмах тих чи інших базових алгоритмічних структур; належності запропонованих алгоритмів тим чи іншим типам; належності змінних заданого алгоритму до аргументів, результатів або проміжних величин;</p> <p>розв'язує: завдання на представлення алгоритмів різними способами; визначенню в алгоритмах аргументів, результатів та проміжних величин;</p> <p>використовує: власні знання і рекомендації учителя для виконання завдань;</p> <p>дотримується правил: опису алгоритмів різними способами з використанням відповідних базових структур.</p>
<p>10. Мови програмування (4 год.) Етапи розв'язування задач з використанням комп'ютера; мова програмування як один із способів описання алгоритму; класифікація мов програмування; середовище програмування; поняття об'єкта; властивості, події та обробники подій; проект і його структура.</p> <p>Практична робота №19. Робота у середовищі програмування</p>	<p>Учень (учениця):</p> <p>називає: різні мови програмування; класи мов програмування; структуру програми; основні можливості середовища програмування;</p> <p>наводить: приклади мов програмування; перелік класів мов програмування; перелік частин та розділів програми; елементів мови програмування; основних можливостей середовища програмування;</p> <p>розпізнає: різні мови програмування та їх призначення; до якого класу відноситься конкретна мова програмування; елементи мови програмування; основні можливості середовища програмування;</p> <p>характеризує: мову програмування як один із способів представлення алгоритму; особливості різних класів мов програмування; особливості мови програмування, що вивчається; особливості частин та окремих розділів програми; особливості окремих можливостей середовища програмування, що вивчається;</p> <p>описує: призначення мов програмування; ознаки різних класів мов програмування; структуру програми;</p> <p>пояснює: призначення окремих мов програмування; ознаки окремих класів мов програмування; призначення окремих частин та розділів програми; призначення середовища програмування та окремих можливостей;</p> <p>формулює: поняття класів мов програмування; назви окремих частин та розділів програми; принципи використання та особливості мови програмування, що вивчається;</p> <p>обґрунтовує: наявність великої кількості різних мов програмування; ознаки окремих мов програмування;</p>

	<p>послідовність частин та розділів програми; необхідність розробки різних середовищ програмування; призначення середовищ програмування; доцільність і переваги використання мови програмування, що вивчається;</p> <p>порівнює: особливості різних мов програмування;</p> <p>класифікує: мови програмування;</p> <p>аналізує: значення доцільного вибору мови програмування для розв'язання поставленої задачі;</p> <p>оцінює: ефективність вибору мови програмування при розв'язуванні поставленої задачі;</p> <p>висловлює судження: щодо необхідності створення різних мов програмування; доцільності використання мови програмування, обраної для навчальних цілей;</p> <p>розв'язує: завдання щодо роботи у середовищі програмування із запропонованою програмою;</p> <p>спост ерігає: за послідовністю виконання програм на комп'ютері;</p> <p>використовує: можливості середовища програмування для виконання програм на комп'ютері;</p> <p>дотримується правил: роботи за комп'ютером; виконання задач на комп'ютері;</p> <p>показує на комп'ютері: послідовність виконання програми у середовищі програмування</p>
<p>11. Лінійні алгоритми (10 год.)</p> <p>Поняття змінної і константи; ім'я та тип; опис стандартних типів; команда присвоювання; оператор присвоювання; арифметичні операції; правила запису арифметичних виразів; стандартні підпрограми, необхідні для запису арифметичних виразів та введення-виведення даних; компоненти для введення та виведення даних; використання текстових файлів для введення та виведення даних; налагодження програм; необхідність і способи описання типів змінних і констант</p> <p>Практична робота №20. Розробка лінійних алгоритмів та їх реалізація у вигляді програм</p> <p>Практична робота №21. Розробка лінійних алгоритмів та їх реалізація у вигляді програм з використанням текстових файлів</p>	<p>Учень (учениця):</p> <p>називає: арифметичні операції; стандартні підпрограми, необхідні для запису арифметичних виразів та введення-виведення даних; команди присвоювання, введення та виведення даних; оператори присвоювання, введення та виведення даних; текстові файли як один із способів введення та виведення даних;</p> <p>наводить: перелік стандартних типів змінних; приклади лінійних алгоритмів, арифметичних виразів, команди та оператора присвоювання, команд та підпрограм введення і виведення даних, текстових файлів для введення та виведення даних;</p> <p>розпізнає: відмінність між іменем та типом змінної; різні стандартні типи змінних; арифметичні операції; арифметичні вирази; стандартні підпрограми; оператор присвоювання; підпрограми введення та виведення даних; текстові файли для введення та виведення даних;</p> <p>характеризує: особливості різних стандартних типів змінних; арифметичні операції; правила запису арифметичних виразів; особливості виклику підпрограм, необхідних для запису арифметичних виразів та введення-виведення даних; призначення команди і оператора присвоювання; призначення команд і підпрограм введення та виведення даних; текстові файли як один із способів введення та виведення даних;</p> <p>описує: стандартні типи змінних; загальний вигляд оператора присвоювання; загальний вигляд підпрограми введення даних; загальний вигляд підпрограми виведення даних; виклик стандартних підпрограм; процес використання текстових файлів для введення та виведення даних;</p> <p>пояснює: поняття змінної; відмінність між іменем та типом змінної; призначення різних стандартних типів змінних; правила запису арифметичних виразів мовою програмування, оператора присвоювання, стандартних підпрограм введення і виведення даних, схеми лінійного алгоритму; принцип виконання команди і оператора присвоювання; принципи використання текстових файлів для введення та виведення даних; результати роботи реалізованого у вигляді програми розробленого алгоритму для запропонованих тестів;</p> <p>формулює: визначення змінної, імені та типу змінної; визначення арифметичного виразу, лінійного алгоритму;</p> <p>обґрунтовує: необхідність використання різних типів змінних для запису алгоритму у вигляді програми; пріоритетність виконання арифметичних операцій; відмінність між поняттями «команда» та «оператор»; особливості виклику підпрограм; переваги використання текстових файлів для введення та виведення даних; значення тестування реалізованих у вигляді програми розроблених лінійних алгоритмів;</p> <p>порівнює: типи змінних; команди та оператори присвоювання, введення та виведення даних; способи введення та виведення даних; результати виконання реалізованого у вигляді програми розробленого лінійного алгоритму для запропонованих тестів;</p> <p>класифікує: арифметичні операції за пріоритетом виконання; стандартні підпрограми за можливостями застосування;</p>

	<p>висловлює судження: щодо смислу введення різних типів змінних; необхідності використання коментарів при розробці алгоритмів у їх тексті; необхідності використання коментарів при введенні початкових даних та при виведенні результуючих даних; необхідності тестування розробленого алгоритму; необхідності використання можливостей середовища програмування під час налагодження лінійної програми;</p> <p>розв'язує: завдання по створенню та тестуванню лінійних алгоритмів з використанням операторів присвоювання, стандартних підпрограм введення і виведення даних, використання текстових файлів для введення та виведення даних, арифметичних виразів;</p> <p>спостерігає: за покроковим виконанням реалізованих у вигляді програми розроблених лінійних алгоритмів; за виконанням програми;</p> <p>використовує: можливості середовища програмування для створення та налагодження лінійних програм; коментарі у тексті програми, при введенні початкових даних та при виведенні результуючих даних;</p> <p>дотримується правил: роботи за комп'ютером; реалізації лінійних алгоритмів у вигляді програм;</p> <p>показує на комп'ютері: покрокове виконання створеної лінійної програми; введення початкових даних; виведення результуючих даних; можливості перегляду значень аргументів, результатів та проміжних величин; виконання лінійної програми для власних розроблених тестів.</p>
<p>12. Елементи алгебри логіки (4 год.)</p> <p>Висловлювання. Істинні та хибні висловлювання. Логічні змінні. Логічні вирази; логічні операції диз'юнкції, кон'юнкції та заперечення; таблиці істинності; прості та складені логічні вирази; обчислення значень логічних виразів</p> <p>Практична робота №22. Побудова логічних виразів та їх обчислення</p>	<p>Учень (учениця):</p> <p>називає: логічні операції диз'юнкції, кон'юнкції та заперечення; види логічних виразів;</p> <p>наводить: приклади простих та складених логічних виразів; приклади виконання логічних операцій;</p> <p>розпізнає: прості та складені логічні вирази; логічні операції;</p> <p>характеризує: особливості простих і складених логічних виразів; особливості виконання логічних операцій;</p> <p>пояснює: відмінність між істинними та хибними висловлюваннями; між простими та складеними логічними виразами; між виконанням логічних операцій;</p> <p>формулює: визначення логічного виразу, простого логічного виразу, складеного логічного виразу;</p> <p>аналізує: відмінність між істинними та хибними висловлюваннями; між простими та складеними логічними виразами;</p> <p>висловлює судження: щодо коректності використання простих або складених логічних виразів;</p> <p>розв'язує: завдання по побудові та обчисленню логічних виразів</p>
<p>13. Алгоритми з розгалуженнями (14 год.)</p> <p>Команда розгалуження; умовний оператор мовою програмування; команда вибору; оператор вибору; вкладені розгалуження; використання компонентів мови програмування для реалізації розгалужень</p> <p>Практична робота №23. Розробка алгоритмів з послідовними розгалуженнями та їх реалізація у вигляді програм</p> <p>Практична робота №24. Розробка алгоритмів з вкладеними розгалуженнями та їх реалізація у вигляді програм</p>	<p>Учень (учениця):</p> <p>називає: види команди та оператора розгалуження; види команди та оператора вибору; компоненти мови програмування для реалізації розгалужень;</p> <p>наводить: приклади простих та вкладених розгалужень;</p> <p>розпізнає: скорочену і повну форму команд і операторів розгалуження; скорочену і повну форму команд і операторів вибору; послідовні та вкладені розгалуження;</p> <p>характеризує: особливості скороченої і повної форм розгалуження, послідовних і вкладених конструкцій розгалуження; особливості двох форм команди вибору;</p> <p>описує: загальний вигляд оператора умовного переходу у повній та скороченій формах; загальний вигляд оператора вибору у повній та скороченій формах;</p> <p>пояснює: відмінність між скороченою і повною формою команди і операції розгалуження; відмінність між скороченою і повною формою команди і операції вибору; особливості використання послідовних і вкладених розгалужень; особливості використання компонентів мови програмування для створення різних видів розгалужень;</p> <p>формулює: означення алгоритму з розгалуженнями;</p> <p>обґрунтовує: необхідність використання і застосування логічних виразів при складанні алгоритмів; необхідність існування двох форм розгалуження і вибору; необхідність коректного використання двох форм розгалуження і вибору, послідовних та вкладених розгалужень; доцільність і значення тестування розроблених алгоритмів з розгалуженнями;</p>

	<p>порівнює: дві різні форми розгалуження та вибору; послідовні та вкладені конструкції розгалужень; результати виконання реалізованого у вигляді програми розробленого алгоритму з розгалуженнями для запропонованих тестів;</p> <p>класифікує: алгоритми з розгалуженнями за двома формами команд розгалуження та вибору, за різними конструкціями розгалужень;</p> <p>аналізує: скороченою та повною формами розгалужень та вибору для заданих прикладів алгоритмів; результати роботи реалізованого у вигляді програми розробленого алгоритму з розгалуженнями;</p> <p>оцінює: результати роботи реалізованого у вигляді програми розробленого алгоритму з розгалуженнями для власних тестів;</p> <p>висловлює судження: щодо скороченої або повної форми розгалуження та вибору, послідовних або вкладених розгалужень у заданих прикладах; необхідності використання можливостей середовища програмування під час налагодження програми з розгалуженнями та використання компонентів мови програмування для створення різних видів розгалужень;</p> <p>розв'язує: завдання по створенню та тестуванню алгоритмів з розгалуженнями з використанням операторів умовного переходу, вибору, різних розгалужених конструкцій;</p> <p>показує на комп'ютері: покрокове виконання створеної програми з розгалуженнями; введення початкових даних; виведення результуючих даних; виконання програми з розгалуженнями для власних розроблених тестів.</p> <p>спостерігає: за покроковим виконанням реалізованих у вигляді програми розроблених алгоритмів з розгалуженнями; за виконанням програми;</p> <p>використовує: можливості середовища програмування для створення та налагодження програм з розгалуженнями; коментарі у тексті програми, при введенні початкових даних та при виведенні результуючих даних;</p> <p>дотримується правил: роботи за комп'ютером; реалізації алгоритмів з розгалуженнями у вигляді програм</p>
<p>14. Алгоритми з повтореннями (20 год.) Команда повторення; оператори циклу; поєднання повторення і розгалуження; числові послідовності; рекурентні послідовності; вкладені цикли; можливості, які забезпечує середовище програмування для роботи з циклічними програмами; використання компонентів мови програмування для реалізації циклів</p> <p>Практична робота №25. Розробка алгоритмів з послідовними повтореннями та їх реалізація у вигляді програм.</p> <p>Практична робота №26. Розробка рекурентних алгоритмів та їх реалізація у вигляді програм</p> <p>Практична робота №27. Розробка алгоритмів з вкладеними повтореннями та їх реалізація у вигляді програм</p>	<p>Учень (учениця):</p> <p>називає: всі різновиди команд і операторів повторення; можливості середовища програмування для роботи з циклічними програмами;</p> <p>наводить: приклади повторень з передумовою, з післяумовою, з параметром; приклади виконання повторень з передумовою, з післяумовою, з параметром; алгоритмів з повтореннями для обчислення значень елементів рекурентних послідовностей, вкладених циклів;</p> <p>розпізнає: циклічні алгоритми з передумовою, з післяумовою, з параметром; алгоритми з повтореннями для обчислення значень елементів рекурентних послідовностей, вкладені цикли;</p> <p>характеризує: особливості запису і виконання циклічних алгоритмів з передумовою, з післяумовою, з параметром; особливості створення і виконання алгоритмів з повтореннями для обчислення значень елементів рекурентних послідовностей; особливості запису і виконання вкладених циклів;</p> <p>описує: загальний вигляд операторів повторення з передумовою, з післяумовою, з параметром; схематичне виконання алгоритмів з повтореннями для обчислення значень елементів рекурентних послідовностей;</p> <p>пояснює: відмінність між виконанням повторень з передумовою, з післяумовою, з параметром; виконання алгоритмів з повтореннями для обчислення значень елементів рекурентних послідовностей та вкладених циклів;</p> <p>формулює: принципи виконання повторень з передумовою, з післяумовою, з параметром; принципи виконання алгоритмів з повтореннями для обчислення значень елементів рекурентних послідовностей; принципи виконання вкладених циклів;</p> <p>обґрунтовує: необхідність та доцільність використання і застосування повторень з передумовою, з післяумовою, з параметром; необхідність використання і застосування алгоритмів з повтореннями для обчислення значень елементів рекурентних послідовностей і вкладених циклів; необхідність і доцільність використання можливостей середовища програмування для налагодження програм з повтореннями; доцільність і значення тестування розроблених алгоритмів з повтореннями;</p>

	<p>порівнює: всі види повторень; послідовні та вкладені повторення; повторення з розгалуженнями та розгалуження з повтореннями;</p> <p>класифікує: алгоритми з повтореннями за різними видами команд повторень;</p> <p>аналізує: відмінність між повтореннями з передумовою, з післяумовою, з параметром для заданих прикладів алгоритмів; відмінність між повтореннями з розгалуженнями та розгалуженнями з повтореннями, з вкладеними циклами, рекурентністю для заданих прикладів алгоритмів; результати роботи реалізованого у вигляді програми розробленого алгоритму з повтореннями;</p> <p>оцінює: результати роботи реалізованого у вигляді програми розробленого алгоритму з повтореннями для власних тестів;</p> <p>висловлює судження: щодо коректності використання у заданих прикладах повторень з передумовою, з післяумовою, з параметром, рекурентних послідовностей, вкладених циклів, повторень з розгалуженнями, розгалужень з повтореннями; необхідності використання можливостей середовища програмування під час налагодження циклічної програми;</p> <p>розв'язує: завдання по створенню та тестуванню алгоритмів з повтореннями з використанням всіх видів операторів повторення, різних циклічних конструкцій;</p> <p>спостерігає: за покроковим виконанням реалізованих у вигляді програми розроблених алгоритмів з повтореннями; за виконанням програми;</p> <p>використовує: можливості середовища програмування для створення та налагодження циклічних програм; коментарі у тексті програми, при введенні початкових даних та при виведенні результуючих даних;</p> <p>дотримується правил: роботи за комп'ютером; реалізації алгоритмів з повтореннями у вигляді програм;</p> <p>показує на комп'ютері: покрокове виконання створеної програми з повтореннями; введення початкових даних; виведення результуючих даних; виконання циклічної програми для власних розроблених тестів</p>
<p>Розв'язування компетентнісних задач. Виконання колективного навчального проекту з дослідження предметної галузі навчального курсу «Інформатика» (8 год)</p>	<p>Розв'язує компетентнісні задачі, що передбачають:</p> <ul style="list-style-type: none"> • змістовий аналіз формулювання задачі; • побудову інформаційної моделі; • розробку стратегії пошуку інформаційних матеріалів; • пошук інформаційних матеріалів; • добір засобів опрацювання даних; • опрацювання даних; • подання результатів розв'язування задачі. <p>Виконує навчальний проект, що передбачає:</p> <ul style="list-style-type: none"> • визначення проблеми, теми та завдань проекту; • розподіл ролей і планування колективної діяльності; • добір засобів опрацювання даних; • добір засобів подання результатів навчального проекту; • розробку стратегії пошуку інформаційних матеріалів; • пошук інформаційних матеріалів і їх аналіз; • створення та опрацювання інформаційної моделі; • аналіз результатів; • підготовку матеріалів; • використання офісних веб-програм та інтернет-середовищ для публікації результатів своєї роботи; • захист проекту
<p>Резерв часу – 4 год.</p>	

(Далі буде)