

**КЛАСИЧНИЙ ПРИВАТНИЙ УНІВЕРСИТЕТ**  
**КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ**

**СИЛАБУС**

**навчальної дисципліни**  
**«ВІЗУАЛЬНЕ ПРОГРАМУВАННЯ»**

**КОНТАКТНА ІНФОРМАЦІЯ ТА ТЕХНІЧНОЇ ДОПОМОГА**  
(включаючи електронну пошту, робочий час / місцезнаходження тощо).

<b>Викладач (-і)</b>	Хрипко Сергій Леонідович
<b>Контактний тел.</b>	+38(061) 228-07-85 (внутр. 224)
<b>E-mail:</b>	ur9qq@ukr.net
<b>Сторінка курсу на сайті підтримки навчальних програм КПУ</b>	<a href="http://www.zhu.edu.ua/cpu_edu/course/view.php?id=4872">http://www.zhu.edu.ua/cpu_edu/course/view.php?id=4872</a>
<b>Консультації</b>	<i>Консультації</i> Очні: за графіком, ауд. 311 (головний корпус КПУ). <i>off-line:</i> за графіком консультацій викладача інформаційно-комунікаційними технологіями: ZOOM

**АНОТАЦІЯ**

Навчальна дисципліна «Візуальне програмування» є нормативною для студентів першого (бакалаврського) рівня вищої освіти галузі знань 01 Освіта/Педагогіка за спеціальністю 014 Середня освіта, освітня програма: Інформатика. Згідно з навчальним планом денної форми навчання вивчення дисципліни заплановано на 5 семестр (3 курс).

Особливістю курсу для спеціальності 014 Середня освіта є його методична спрямованість на підготовку майбутнього вчителя інформатики до викладання основ алгоритмізації та програмування з використанням візуальних мов і блокових середовищ. Дисципліна орієнтована на розвиток умінь створювати інтерактивні навчальні моделі, демонстраційні приклади й тренажери, що сприяють формуванню алгоритмічного мислення в учнів та забезпечують наочність під час пояснення тем шкільного курсу інформатики.

Зміст дисципліни побудовано навколо опанування принципів візуальних мов програмування у середовищі Google Blockly (блокове програмування), зокрема роботи з базовими категоріями блоків та їх застосуванням для побудови алгоритмів і навчальних проєктів.

Освітній процес з дисципліни здійснюється за такими формами: навчальні заняття; самостійна робота; контрольні заходи. Видами навчальних занять згідно з навчальним планом є лекції; практичні заняття, а також консультації.

Практичні заняття передбачають розробку прикладних проєктів освітнього спрямування на базі Blockly: інтерактивні вправи з арифметики та логіки, тренажери з умовами й циклами, мінісимуляції, демонстрації обробки текстів і масивів, а також алгоритмічні візуалізатори, де колір і складені інтерфейси використовуються для показу станів і кроків виконання алгоритму.

У результаті студенти формують набір готових матеріалів для уроків, факультативів і гурткової роботи.

Самостійна робота проводиться під час аудиторних занять та в час, вільний від обов'язкових навчальних занять, без участі викладача шляхом самостійного опрацювання лекційного матеріалу, виконання індивідуальних завдань до кожної теми (побудова програм із блоків, тестування, вдосконалення проєктів, підготовка пояснювальних матеріалів для учнів). Повний курс лекційного матеріалу та методичні рекомендації до виконання індивідуальних домашніх завдань розміщено на сторінках дисципліни сайту підтримки навчальних програм університету.

Консультації призначені для роз'яснення студентам теоретичних або практичних питань, пов'язаних із побудовою алгоритмів у Blockly, типових помилок у логіці та керувальних конструкціях, а також із проєктуванням наочних демонстрацій.

Засвоєння навчального матеріалу перевіряється за допомогою поточного контролю, який здійснюється на практичних заняттях у формі усних відповідей, самостійних робіт, перевірки виконання індивідуальних завдань і захисту практичних проєктів. Для визначення результатів модульного та підсумкового контролю використовується система накопичення балів, яка стимулює систематичну роботу студента протягом семестру.

Підсумковий (семестровий) контроль після завершення 3 семестру здійснюється у формі екзамену.

## ФОРМАТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**Загальна кількість годин – 120 год., у т. ч. 48 годин аудиторних занять і 72 годин самостійної роботи студента. Кількість кредитів ECTS – 4.**

Всього кредитів	Всього годин	Аудиторних годин	У тому числі			Сам. робота
			Лекц.	Лабор.	Семін. (практ.)	
4	120	48	24	-	24	72

## ОЗНАКИ ДИСЦИПЛІНИ

Курс (рік навчання)	Семестр	Цикл підготовки	Нормативна/вибіркова
3	5	професійна	нормативна

## МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### Мета навчальної дисципліни

Метою навчальної дисципліни є формування у майбутніх учителів інформатики практичних навичок розробки алгоритмів і навчальних програм засобами візуальних мов програмування у середовищі Google Blockly, а також опанування методів візуалізації виконання базових алгоритмічних конструкцій (розгалуження, цикли, робота зі змінними, масивами/списками) і створення складених інтерактивних інтерфейсів для демонстрації навчального матеріалу та розв'язування типових задач шкільного курсу інформатики.

### Завдання навчальної дисципліни

- вивчення принципів візуальних мов програмування та особливостей блокового подання алгоритмів у Google Blockly;
- засвоєння можливостей базових каталогів Blockly: Math, Logic, Control для побудови обчислювальних, логічних і керувальних конструкцій;
- оволодіння прийомами проєктування складених інтерфейсів і інтерактивних навчальних сценаріїв (ввід/вивід даних, керування виконанням, демонстрація кроків алгоритму);

- набуття навичок роботи з текстами та символічними змінними, а також з масивами/списками для підготовки даних і реалізації алгоритмів;
- формування вмінь застосовувати колірну палітру як засіб наочності (візуальне кодування станів, виділення елементів, пояснення кроків алгоритму);
- розвиток навичок декомпозиції задач через створення функцій і підпрограм у Blockly для повторного використання блоків і структурування навчальних проєктів.

## **ЗАПЛАНОВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ**

### **У результаті вивчення дисципліни студенти повинні знати:**

- основні поняття та компоненти візуальних мов програмування, принципи побудови програм у Google Blockly (робоче поле, блоки, з'єднання, типи даних);
- призначення та можливості базових каталогів Blockly: Math, Logic, Control, а також правила комбінування блоків для побудови коректних алгоритмів;
- механізми керування виконанням і взаємодії користувача з проєктом у візуальному середовищі (введення/виведення, керувальні елементи, запуск/зупинка, покроковість);
- способи візуального подання алгоритмів і даних, зокрема роботу з текстами, символічними змінними, масивами/списками та прийоми відображення їх станів під час виконання;
- принципи використання колірної палітри для наочності (маркування станів, виділення елементів, пояснення кроків алгоритму);
- підходи до декомпозиції та повторного використання рішень через функції і підпрограми у блоковому програмуванні.

### **Після вивчення дисципліни студенти повинні вміти:**

- створювати та налагоджувати алгоритми у Google Blockly із використанням блоків Math/Logic/Control, формувати коректні обчислювальні й логічні вирази, будувати розгалуження та цикли;
- реалізовувати введення та виведення даних у навчальних проєктах, організовувати взаємодію користувача з інтерфейсом (поля, повідомлення, керувальні елементи);
- розробляти інтерактивні візуалізації для демонстрації роботи алгоритмів (покрокове виконання, підсвічування, зміна станів, візуальне кодування кольором);
- застосовувати роботу з текстами та масивами/списками для розв'язування типових навчальних задач і підготовки даних до візуалізації;
- проєктувати складені інтерфейси (композиція елементів, логіка керування, сценарії взаємодії) та створювати навчальні мініпроєкти (тренажери, тести, симуляції);
- структурувати програми шляхом декомпозиції на функції/підпрограми, забезпечувати повторне використання блоків та читабельність алгоритмічних рішень.

Відповідно до освітньо-професійної програми підготовки бакалавра галузі знань 01 Освіта/Педагогіка за спеціальністю 014 Середня освіта, освітня програма: Інформатика вивчення дисципліни «Візуальне програмування» сприяє формуванню **компетентностей та програмних результатів навчання:**

### **Інтегральна компетентність:**

Здатність розв'язувати складні спеціалізовані задачі у галузі середньої освіти, що передбачає застосування теоретичних знань і практичних умінь з наук предметної спеціальності, педагогіки, психології, теорії та методики навчання і характеризується комплексністю та невизначеністю умов організації освітнього процесу в закладах середньої освіти.

### **Загальні компетентності:**

ЗК 3. Інформаційно-комунікаційна компетентність. Здатність ефективно використовувати сучасні ІКТ для пошуку, обробки, критичної оцінки та поширення інформації в освітньому та соціальному просторах.

ЗК 7. Здатність до креативності та інновацій. Здатність генерувати нові ідеї, виявляти

гнучкість у професійних ситуаціях та застосовувати творчий підхід до розв'язання практичних проблем.

### Спеціальні (фахові) компетенції:

СК 3. Програмно-технологічна компетентність. Здатність розробляти програмні продукти, використовуючи сучасні мови програмування, знання архітектури ПК, операційних систем та мереж.

### Програмні результати навчання:

РН 3. Використовує сучасні мови, методи та технології програмування для розв'язання практичних завдань і створення програмних продуктів.

РН 7. Буде інформаційні та математичні моделі об'єктів, процесів і явищ; використовує бази даних та технології візуалізації для опрацювання і представлення даних.

## ПЛАН КУРСУ

Назва змістових модулів та тем	Лекц.	Лаб.	Практ. (сем.)	Завдання для самостійної роботи
<b>Змістовий модуль № 1. Введення у візуальні мови програмування</b>				
Тема №1. Введення у візуальні мови програмування	4	-	4	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: • Поняття візуальних мов програмування та їх відмінності від текстових мов (переваги, обмеження, сфери застосування). • Основні елементи середовища Google Blockly: робоча область, панель (каталог) блоків, типи блоків, правила їх з'єднання та структура блокової програми. 3. Виконання індивідуального завдання № 1. 4. Виконання тестового завдання на сайті підтримки навчальних програм.
Тема №2. Каталог Math візуального середовища розробки Google Blockly	4	-	2	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: • Склад і призначення блоків каталогу Math у Google Blockly (числові значення, арифметичні операції, округлення, випадкові числа). • Правила побудови та перевірки коректності математичних виразів у Blockly: пріоритет операцій, вкладені блоки, перетворення типів і типові помилки під час обчислень. 3. Виконання індивідуального завдання № 2. 4. Виконання тестового завдання на сайті підтримки навчальних програм.
Тема №3. Каталог Logic	2	-	2	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: • Склад і призначення блоків каталогу <b>Logic</b> у Google Blockly (логічні значення, порівняння, логічні операції AND/OR/NOT, перевірка істинності умов). • Принципи побудови складених логічних виразів у Blockly: комбінування порівнянь і логічних операцій, вкладеність, пріоритети, типові помилки (зайві

Назва змістових модулів та тем	Лекц.	Лаб.	Практ. (сем.)	Завдання для самостійної роботи
				заперечення, некоректні порівняння, “завжди істинні/хибні” умови). 3. Виконання індивідуального завдання № 3. 4. Виконання тестового завдання на сайті підтримки навчальних програм.
Тема №4. Каталог Control	2		4	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: • Склад і призначення блоків каталогу Control у Google Blockly: розгалуження (if/ else), цикли (repeat, while/until, for), вкладені керувальні конструкції та керування потоком виконання. • Правила проєктування алгоритмів із керувальними конструкціями: побудова умов для розгалужень і циклів, уникнення нескінченних циклів, організація вкладених структур, типові помилки та прийоми налагодження виконання блокових програм. 3. Виконання індивідуального завдання № 4. 4. Виконання тестового завдання на сайті підтримки навчальних програм. 5. Підготовка до модульного контролю за темами 1-4.
<b>Змістовий модуль № 2. Робота з даними, візуальне кодування та модульність</b>				
Тема №5. Робота з текстами та символічними змінними	4	-	2	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: • Основні блоки для роботи з текстами та символічними змінними у Google Blockly: створення рядків, конкатенація, довжина, пошук/витяг фрагментів, порівняння текстів, перетворення регістру. • Особливості введення/виведення та обробки текстових даних у блокових програмах: коректне використання змінних, перетворення типів (текст ↔ число), типові помилки (зайві пробіли, некоректні порівняння, змішування типів). 3. Виконання індивідуального завдання № 5. 4. Виконання тестового завдання на сайті підтримки навчальних програм. 4. Виконання тестового завдання на сайті підтримки навчальних програм.
Тема №6. Елементи для роботи з масивами	4	-	2	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: • Основні блоки для роботи з масивами/списками у Google Blockly: створення списку, додавання/вставка/видалення елементів, отримання та заміна елемента за індексом, визначення довжини списку, пошук елемента. • Принципи опрацювання масивів у блокових алгоритмах: індексація, перебір елементів у циклах, вкладені структури (список у списку), типові помилки (вихід за межі індексу, плутанина “перший/останній”, зміна списку під час перебору) та способи їх уникнення. 3. Виконання індивідуального завдання № 6.

Назва змістових модулів та тем	Лекц.	Лаб.	Практ. (сем.)	Завдання для самостійної роботи
				4. Виконання тестового завдання на сайті підтримки навчальних програм.
Тема №7. Робота з колірною палітрою	2	-	4	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: <ul style="list-style-type: none"> <li>• Призначення та способи використання кольору у візуальних проєктах (у т.ч. Blockly): кольорове кодування станів, підсвічування активних елементів, відображення результатів і помилок, підвищення наочності демонстрацій.</li> <li>• Основи роботи з колірною палітрою: подання кольорів (RGB/HEX), вибір контрастних поєднань, правила читабельності та доступності (контраст, уникнення “злиття” кольорів, обмеження для людей із порушеннями кольоросприйняття), типові помилки та рекомендації для навчальних візуалізацій.</li> </ul> 3. Виконання індивідуального завдання № 7. 4. Виконання тестового завдання на сайті підтримки навчальних програм.
Тема №8. Створення функцій і підпрограм	2		4	1. Опрацювання лекційного матеріалу, 2. Самостійне опрацювання теоретичних питань: <ul style="list-style-type: none"> <li>• Поняття функцій і підпрограм у Google Blockly: призначення, структура, область видимості змінних, повторне використання блоків, відмінність між процедурою та функцією (з/без повернення значення).</li> <li>• Принципи <b>декомпозиції</b> алгоритмічних задач: виділення підзадач, проектування параметрів і результатів, уникнення дублювання коду, типові помилки (надто “великі” функції, зайві параметри, приховані залежності) та правила оформлення бібліотеки підпрограм для навчальних проєктів.</li> </ul> 3. Виконання індивідуального завдання № 8. 4. Виконання тестового завдання на сайті підтримки навчальних програм. 5. Підготовка до модульного контролю за темами 5-8.

## ФОРМИ КОНТРОЛЮ ТА КРИТЕРІЇ ОЦІНЮВАННЯ

У процесі вивчення навчальної дисципліни «Сучасні мови програмування» використовуються наступні види контролю:

1. Поточний контроль – здійснюється протягом семестру шляхом перевірки виконання тестових завдань, виконання лабораторних робіт, модульних контрольних робіт тощо. За змістом він включає перевірку ступеню засвоєння студентом навчального матеріалу, який охоплюється темою лекційного заняття, уміння самостійно опрацьовувати навчально-методичну літературу, здатність осмислювати зміст теми, уміння публічно та письмово представити певний матеріал, а також виконання завдань самостійної роботи.

2. Підсумковий семестровий контроль – здійснюється у формі письмового екзамену відповідно до графіка освітнього процесу шляхом визначення ступеню засвоєння студентом навчальної дисципліни за результатами виконання обов’язкових завдання поточного

(модульного) контролю.

Для оцінювання студентів використовується система накопичування балів. Згідно з «Положенням про організацію освітнього процесу в КПУ» підсумкова оцінка з дисципліни виставляється за 100-бальною шкалою з наступним переведенням у національну шкалу та шкалу ECTS.

Бали нараховуються за виконання завдань аудиторної роботи, практичних, лабораторних робіт, контрольних (модульних) завдань, тестів.

Результати поточного контролю здобувачів вищої освіти є складовими елементами підсумкової оцінки з дисципліни.

Оцінка рівня роботи студента протягом семестру під час навчальних занять та самостійної роботи здійснюється у межах 80 балів. Вага екзамену у підсумковій оцінці складає 20 балів.

## РОЗПОДІЛ БАЛІВ ПОТОЧНОГО ТА ПІДСУМКОВОГО (СЕМЕСТРОВОГО) ОЦІНЮВАННЯ

Поточне оцінювання під час навчальних занять та самостійної роботи								Екзамен	Сума
Змістовий модуль 1				Змістовий модуль 2					
T1	T2	T3	T4	T4	T5	T6	T4	20	100
10	10	10	10	10	10	10	10		

### Критерії оцінювання

Для оцінювання відповідей студентів з навчальної дисципліни «Сучасні мови програмування» використовуються наступні **критерії**:

– рівню «відмінно» відповідає теоретично правильна і вичерпна відповідь на поставлене питання, у якій студент показав всебічне системне знання програмного матеріалу; засвоєння основної та додаткової літератури; чітке володіння понятійним апаратом, методами, методиками та інструментами, вивчення яких передбачене програмою дисципліни; уміння використовувати знання для аналізу життя економічного суспільства та аргументувати своє ставлення до відповідних категорій, закономірностей, випадковостей, суспільних явищ і процесів;

– рівню «добре» відповідає теоретично правильна, але не вичерпна відповідь на поставлене запитання, в цілому повне знання програмного матеріалу, успішне виконання запропонованого завдання і засвоєння матеріалу основної літератури;

– рівню «задовільно» відповідає у цілому правильна відповідь на поставлене питання, в якій студент показав достатній рівень знань з основного програмного матеріалу дисципліни, але не зміг переконливо аргументувати свою відповідь, помилився у використанні понятійного апарату, показав недостатні знання рекомендованої літератури;

– рівню «незадовільно» відповідає неправильна або неповна відповідь на запитання, у якій студент продемонстрував значні прогалини у знаннях з основного програмного матеріалу; ухилився від аргументувань; показав незадовільні знання понятійного апарату і спеціальної літератури чи взагалі нічого не відповів.

## Шкала оцінювання: 100-бальна, національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90-100	A	відмінно	зараховано
82-89	B	добре	
75-81	C		
67-74	D	задовільно	
60-66	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

### РЕКОМЕНДОВАНА ЛІТЕРАТУРА

#### Підручники та навчальні посібники

1. Bers M. U., Resnick M.. The Official ScratchJr Book: Help Your Kids Learn to Code. – San Francisco : No Starch Press, 2015. – 160 p.
2. Marji M.. Learn to Program with Scratch: A Visual Introduction to Programming with Games, Art, Science, and Math. – San Francisco : No Starch Press, 2014. – 288 p.
3. Woodcock J.. Coding Projects in Scratch: A Step-by-Step Visual Guide to Coding Your Own Animations, Games, Simulations, and More! – London : DK, 2019. – 224 p.
4. Sande W. D., Sande C.. Hello World!: Computer Programming for Kids and Other Beginners. – Greenwich : Manning Publications, 2009. – 432 p.

#### Статті у наукових фахових виданнях

1. Resnick M., Maloney J., Monroy-Hernández A. та ін. Scratch: programming for all // Communications of the ACM. – 2009. – Vol. 52, no. 11. – P. 60–67. – DOI: 10.1145/1592761.1592779.
2. Bau D., Gray J., Kelleher C. та ін. Learnable programming: blocks and beyond // Communications of the ACM. – 2017. – Vol. 60, no. 6. – P. 72–80. – DOI: 10.1145/3015455.
3. Lin Y., Weintrop D.. The landscape of block-based programming: Characteristics of block-based environments and how they support the transition to text-based programming // Journal of Computer Languages. – 2021. – Vol. 67. – Art. 101075. – DOI: 10.1016/j.cola.2021.101075.
4. Scherer R., Siddiq F., Sánchez Viveros B.. A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions // Computers in Human Behavior. – 2020. – Vol. 109. – Art. 106349. – DOI: 10.1016/j.chb.2020.106349.

### ІНФОРМАЦІЙНІ РЕСУРСИ В ІНТЕРНЕТІ

#### Програмне забезпечення та документація

1. Blockly: офіційна документація та довідник API [Електронний ресурс]. <https://developers.google.com/blockly> – Дата звернення: 13.02.2026.
2. Blockly Developer Tools (створення власних блоків) [Електронний ресурс]. <https://developers.google.com/blockly/guides/create-custom-blocks/blockly-developer-tools> – Дата звернення: 13.02.2026.

#### Електронні підручники та освітні платформи

3. ScratchEd: Creative Computing Curriculum Guide [Електронний ресурс].

<https://scratched.gse.harvard.edu/resources/scratch-curriculum-guide.html> – Дата звернення: 13.02.2026.

4. Prometheus: платформа онлайн-курсів (курси з ІТ/цифрових навичок) [Електронний ресурс].

<https://prometheus.org.ua/> – Дата звернення: 13.02.2026.

5. Blockly Games: тренажери блокового програмування [Електронний ресурс].

<https://blockly.games/> – Дата звернення: 13.02.2026.

6. Code.org App Lab: середовище для створення застосунків (blocks/text) [Електронний ресурс].

<https://code.org/en-US/tools/app-lab> – Дата звернення: 13.02.2026