

Вибіркові модулі. Інформатика 10-11 клас

Методичні рекомендації



Вибіркові модулі.
Інформатика 10-11 клас

Методичні рекомендації

В 41 Вибіркові модулі. Інформатика 10-11 клас : методичні рекомендації / Укладачі М. О. Величко, С. Д. Настусенко, Б. М. Теренчин, Я. В. Левицький, Т. О. Попруга. За ред. І. В. Лупан. – Кропивницький, ФОП Піскова М. А., 2023. – 180 с.

У збірнику наведено рекомендації до викладання вибірових модулів «Комп'ютерна анімація», «Інформаційна безпека», «Веб-технології», «Креативне програмування» та курсу за вибором «Програмування на Java», зокрема, орієнтовні календарно-тематичні плани, переліки літератури та інтернет-джерел для вчителів та учнів, огляди доступного програмного забезпечення, приклади практичних завдань та питання до тестового контролю.

Рекомендації укладено групою магістрантів Центральноукраїнського державного університету імені Володимира Винниченка спеціальності «014 Середня освіта» освітньої програми «Математика, інформатика та економіка». Вони будуть корисними як для студентів, так і для вчителів інформатики закладів загальної середньої освіти, а також для учнів, які опановують відповідні розділи інформатики самостійно.

Рецензенти:

Чередніченко Н. Ю. – директор комунального закладу "Лицей "НАУКОВИЙ" міської ради міста Кропивницький", доктор педагогічних наук;

Скрипка Г.В. – кандидат пед. наук, зав. кафедри інформаційно-комунікаційних технологій та безпечного освітнього середовища КЗ «КОІППО імені Василя Сухомлинського»;

Резіна О.В. – кандидат пед. наук, доцент кафедри інформатики та інформаційних технологій Центральноукраїнського державного університету імені Володимира Винниченка.

Рекомендовано методичною радою Центральноукраїнського державного університету імені Володимира Винниченка (протокол №1 від 29.11. 2023)

Зміст

Вибірковий модуль «Комп'ютерна анімація» (35 год.) .. 5

Календарно – тематичне планування:	5
Список рекомендованої літератури (для вчителя):	7
Онлайн джерела для вчителів:	9
Список рекомендованої літератури для учнів	9
Програмне забезпечення для створення комп'ютерних анімацій.....	10
Програми для створення 3D анімацій:	12
Онлайн-сервіси для створення комп'ютерної анімації:	13
Практичні завдання до уроків	14
Підсумковий тест з вибіркового модуля «Комп'ютерна анімація»	41

Вибірковий модуль «Інформаційна безпека» (17 годин)

.....	45
Календарно-тематичний план	45
Список рекомендованої літератури для вчителя.....	46
Інтернет джерела:	48
Список рекомендованої літератури для учнів	48
Інтернет джерела:	49
Програмне забезпечення:.....	49
Завдання до практичних занять	53
Підсумковий тест з вибіркового модуля «Інформаційна безпека»	72

Вибірковий модуль «Веб-технології» (35 год.)..... 77

Календарно-тематичний план	77
Список рекомендованої літератури (для вчителя):.....	79
Список рекомендованої літератури для учнів.....	80
Програмне забезпечення для створення комп'ютерних анімацій.....	81
Практичні завдання до уроків	81
Тести до практичних робіт	92

Вибірковий модуль «Креативне програмування» (35 годин)

.....	101
Календарно-тематичний план	101
Література для вчителя.....	103

Література для учнів.....	104
Програмне забезпечення:	104
Практичні завдання до уроків:	107
Завдання для тестування.	141
Курс за вибором «Програмування на Java»	166
Розподіл годин на вивчення курсу (1 семестр – 70 годин)	166
Список рекомендованої літератури для вчителя:	167
Список рекомендованої літератури та електронних джерел для учнів:	168
Програмне забезпечення:	168
Приклади практичних завдань	172

Вибірковий модуль «Комп'ютерна анімація» (35 год.)

Укладач: М.О.Величко

Календарно – тематичне планування:

Календарно-тематичне планування складене відповідно до чинної програми «Інформатика 10 – 11 класи (рівень стандарту). Програма для загальноосвітніх навчальних закладів». Затверджено Наказом Міністерства освіти і Науки України від 23.10.2017 № 1407.

Посилання:

<https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-10-11-klas/2018-2019/informatika-standart-10-11.docx>

№ уроку	Зміст навчального матеріалу	Домашнє завдання
1.	Анімація. Види анімації. Комп'ютерна анімація.	
2.	Порівняння растрової та векторної анімації.	
3.	Тривимірне моделювання і анімація.	
4.	Програмні середовища для створення анімації. Формати файлів комп'ютерної анімації.	
5.	<i>Практична робота №1:</i> «Пошук та перегляд файлів з анімацією різних видів».	
6.	Етапи створення анімованого зображення. Приклади застосування анімації.	
7.	<i>Практична робота №2:</i> «Створення простого анімованого зображення»	
8.	Основи растрової графіки. Використання фото та кліпартів. Підготовка малюнків для створення анімації.	
9.	Призначення та основні функції редактора GIF-анімацій.	
10.	Покадрова анімація. Анімація на основі фотоколажу. Експортування анімації.	

№ уроку	Зміст навчального матеріалу	Домашнє завдання
11.	<i>Практична робота №3: «Створення GIF-анімацій».</i>	
12.	Статичні та динамічні зображення. Використання шарів. Анімація декількох об'єктів.	
13.	Налаштування швидкості відтворення анімації. <i>Практична робота №4: «Анімація декількох об'єктів. Налаштування швидкості відтворення анімації».</i>	
14.	Анімація руху об'єктів.	
15.	Анімація зіткнення декількох об'єктів.	
16.	Ефекти анімації обертання об'єктів. <i>Практична робота №5: «Ефекти анімацій, рух об'єктів в презентаціях».</i>	
17.	Інструменти трансформації і деформації.	
18.	Анімація кольору. Градієнтні заливки, текстури. Ефекти анімації з колірними переходами, анімація прозорості.	
19.	Використання візуальних ефектів в анімації. <i>Практична робота №6: «Створення анімацій з використання різних ефектів».</i>	
20.	Анімація тексту. <i>Практична робота №7: «Анімація тексту».</i>	
21.	Моделювання явищ та процесів засобами анімації. <i>Практична робота №8: «Створення анімованих моделей явищ та процесів».</i>	
22.	Створення анімованого банера для веб-сторінок. Зміна тексту та зображень в анімованому банері.	
23.	<i>Практична робота №9: «Створення анімованого банера для веб-сторінок».</i>	
24.	<i>Практична робота №10: «Розробка короткого мультфільму з використання різних ефектів»</i>	

№ уроку	Зміст навчального матеріалу	Домашнє завдання
25.	Основи векторної графіки. Призначення та основні функції редактора векторної графіки.	
26.	Створення та редагування векторних зображень. <i>Практична робота №11: «Створення власних векторних зображень їх редагування»</i>	
27.	Векторна анімація. Види векторної анімації: покадрова, руху, форми, з використанням криволінійних шляхів.	
28.	<i>Практична робота №12: «Створення векторної анімації».</i>	
29.	Налаштування швидкості відтворення анімації. Прискорення та уповільнення.	
30.	Використання шарів для створення анімації, змінення їх властивостей.	
31.	Анімація руху одного та кількох об'єктів. Анімація обертання.	
32.	<i>Практична робота №13: «Редагування початкових файлів анімації. Експорт анімації».</i>	
33.	Інструменти трансформації і деформації. Анімація кольору.	
34.	Створення керованої анімації. <i>Практична робота №14: «Створення керованої анімації».</i>	
35.	<i>Практична робота №15: «Розробка короткого анімаційного ролика з використанням створеної графіки»</i>	

Список рекомендованої літератури (для вчителя):

1. Вільямс Р. Е. Анімація: Посібник з виживання. (методи, принципи та інструменти для класичної, цифрової, стоп-кадрової анімації) / Пер. з англ. К. : Вид-во: ArtHuss, 2019. 392 с.

2. Євсєєв О.С. Комп'ютерна анімація: навчальний посібник / О.С. Євсєєв. Х. : Вид. ХНЕУ ім. С. Кузнеця, 2014. 155 с.
3. Мельник О.С. Комп'ютерна анімація та 3D-моделювання: Навчальний посібник / Укладач: О.С. Мельник. Умань : УДПУ ім. Павла Тичини, 2018. 141 с.
4. Селбі Е. Анімація. / Пер. з англ. К. : Вид-во: ArtHuss, 2019. 216 с.
5. Parent, R. (2012). Computer Animation (3rd ed.). Elsevier Science. Retrieved from <https://www.perlego.com/book/1814049/computer-animation-algorithms-and-techniques-pdf> (Original work published 2012)
6. Комп'ютерна графіка : навчально-методичний посібник / [укладач Т. Г. Четверикова]. – Луцьк, 2016. 110 с.
7. Комп'ютерна графіка : навчальний посібник : в 2-х кн. 2. для студентів спеціальності 151 «Автоматизація та комп'ютерноінтегровані технології» / Укладачі : Тотосько О.В., Микитишин А.Г., Стухляк П.Д. Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2017. 304 с.
8. Інформатика (профільний рівень): підруч. для 10 кл. закл. загал. серед. освіти/ В. Д. Руденко, Н. В. Речич, В. О. Потієнко. – Харків : Вид-во «Ранок», 2018. 255 с.
9. Інформатика: графічний дизайн(вибірковий модуль для учнів 10 – 11 класів, рівень стандарту) /В.О. Потієнко. Харків: Вид-во «Ранок», 2020. 160 с.
10. Пічугін М.Ф., Канкін І.О., Воротніков В.В. Комп'ютерна графіка: навч. посібник для вищих навч. закладів. К. : Центр навчальної літератури, 2019. 346 с.
11. Обідник Д.Т. Комп'ютерна анімація: [навчальний посібник] / Д.Т. Обідник. Вінниця: ВНТУ, 2004. 124 с.
12. Веселовська Г.В. Основи комп'ютерної графіки: навч. посібник / Г. В. Веселовська, В. Є. Ходаков, В. М.

Веселовський. К.: Центр навчальної літератури, 2004. 392 с.

13. Васильєв В. Е. Компьютерная графика : Учеб. Пособие / В. Е. Васильєв, А. В. Морозов. – СПб. : СЗТУ, 2005. 101 с.
14. Пасічна Л.І., Пасічний М.О. Сто вправ до курсу «Основи комп'ютерної графіки»: Практикум. Черкаси: ЧНУ ім. Б. Хмельницького, 2012. 44 с.

Онлайн джерела для вчителів:

- ✓ Анімація. Види анімації. Комп'ютерна анімація. – <http://axiom.at.ua/index/informatika/0-39>
- ✓ Модуль «Комп'ютерна анімація»: <https://vsimpptx.com/author/informatika-10-11-klas-modul-kompyuterna-animaciya>
- ✓ Модуль «Комп'ютерна анімація»: <https://sites.google.com/site/nmsurokinformatyky/uroki/10-klas/vibirkovij-modul-komp-uterna-animacia>
- ✓ Synfig. Вступ до комп'ютерної анімації: <https://www.youtube.com/watch?v=cm0TY0V0cZM&list=PLmLnIAKOxguh2KQ7vGTSuF7fRg3Y3p1m>
- ✓ Міні курс «Мауа для початківців»: <https://www.youtube.com/watch?v=nMGZu6ahc28&list=PLrsq-o51mMFFGTwUBMBWZjYHWCJjuxFfx>

Список рекомендованої літератури для учнів

1. Інформатика (профільний рівень): підруч. для 10 кл. закл. загал. серед. освіти/ В. Д. Руденко, Н. В. Речич, В. О. Потієнко. – Харків : Вид-во «Ранок», 2018. 255 с.
2. Комп'ютерна графіка : навчально-методичний посібник / [укл. Т. Г. Четверикова] – Луцьк, 2016. – 110 с.
3. Євсєєв О.С. Комп'ютерна анімація: навчальний посібник / О.С. Євсєєв. – Х. : Вид.ХНЕУ ім. С. Кузнеця, 2014. 155.

4. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закл. / [О.О. Бондаренко, В.В. Ластовецький, О.П. Пилипчук, Є.А. Шестопапов]. – Харків : Вид-во «Ранок», 2022. – 237 с.
5. Інформатика : підруч. для 6-го кл. закл. заг. серед. освіти / Й. Ривкінд [та ін.]. — Київ : Генеза, 2019. 128 с.

Відеоуроки, які допоможуть учням створювати анімації в різних графічних редакторах:

- ✓ Анімація в растровому графічному редакторі GIMP – <https://youtu.be/jZW3BXA3FI>
- ✓ Анімація в растровому графічному редакторі Photoshop – <https://youtu.be/ENj3bOUzHkw>
- ✓ Анімація в Blender – <https://youtu.be/wlh9i0yKI28>
- ✓ Synfig. Вступ до комп'ютерної анімації – <https://www.youtube.com/watch?v=cm0TY0V0cZM&list=PLmLnIAKOxguh2KQ7vGTSuF7fRgb3Y3p1m>
- ✓ Міні курс «Мауа для початківців» – <https://www.youtube.com/watch?v=nMGZu6ahc28&list=P Lrsq-o51mMFFGTwUBMBWZjYHWCJjuxFfx>

Програмне забезпечення для створення комп'ютерних анімацій

- 1) **GIMP** – редактор растрової графіки. Дозволяє створювати растрову анімацію.
Розробник: The GIMP Team
Лицензія: Безкоштовне програмне забезпечення
Операційна система: Windows 7/8/10/11
Сайт розробника: <https://www.gimp.org/>
- 2) **PivotStickfigureAnimator** – безкоштовний додаток для створення двовимірної векторної анімації, що не вимагає особливих навичок малювання і має простий і зрозумілий інтерфейс. Процес створення заснований на переміщенні фігур, складених з ліній та кіл (скелетна анімація), а також спрайтів. Програма дозволяє зберігати анімації у форматах GIF та AVI (без звуку).

Розробник: Пітер Боун

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: Windows 7/8/10/11

Сайт розробника: <https://pivotanimator.net/>

- 3) **Stykz** – це безкоштовна програма для створення 2D-анімації, яка дозволяє створювати та редагувати кадри шляхом руху об'єктів.

Розробник: Sons of Thunder Software

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: Windows 7/8/10

Сайт розробника: <http://www.stykz.net/>

- 4) **Synfig Studio** – це інструмент для створення 2D анімації, який здатний працювати як з векторною, так і з растровою графікою, і який дозволить створювати анімацію швидше та споживає менше ресурсів.

Розробник: Synfig Studio

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: Linux, Mac OS X, Windows

Сайт розробника: <https://www.synfig.org/>

- 5) **UnFREEz** – програма, яка на основі зображень, збережених окремими файлами GIF, створює єдиний анімований GIF

Розробник: WhitSoft Development

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: Windows 7/8/10

Сайт розробника: <http://www.whitsoftdev.com/unfreez/>

- 6) **Adobe Photoshop** – редактор у якому можна створювати растрову анімацію.

- 7) Розробник: Adobe Systems

Ліцензія: Власницьке програмне забезпечення (є безкоштовна пробна версія)

Операційна система: Windows, Mac OS X, iOS, Android

Сайт розробника: <https://www.adobe.com/ua/>

- 8) **Macromedia Flash** – мультимедійна та програмна платформа використовувана для авторської розробки векторної графіки, анімації, ігор.

Розробник: Adobe Systems

Ліцензія: Власницьке програмне забезпечення

Операційна система: Windows 7/8/10

Сайт розробника: <https://www.softportal.com/>

- 9) **Adobe Animate** – програма для створення Flash-ігор, анімаційних фільмів, кінетичної типографіки, мультфільмів, анімованих GIF-файлів тощо. Працює як з векторною, так і з растровою графікою.

Розробник: Adobe Systems

Ліцензія: умовно-безкоштовне (пробний період)

Операційна система: Windows, macOS

Сайт розробника: <https://www.adobe.com/ua/>

- 10) **Pencil2D** – безкоштовна програма для створення 2D-анімації, яка має простий інтерфейс та легко використовується. Вона має вбудовані інструменти для малювання, створення об'єктів та руху.

Розробник: Matt Chang, et al

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: Linux, Windows, Mac OS

Сайт розробника: <https://www.pencil2d.org/>

- 11) **TupiTube** – безкоштовна програма для створення 2D-анімації, яка має простий інтерфейс та великий вибір інструментів для створення анімації. Діти можуть створювати і спільно використовувати малюнки і анімацію легко або зі своїх мобільних телефонів або планшетів.

Розробник: Густав Гонсалес

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: Windows, Mac OS X

Сайт розробника: <https://www.tupitube.com/downloads/>

Програми для створення 3D анімацій:

- 12) **Blender** – одна з найкращих анімаційних програм із відкритим кодом, а також високоякісне програмне забезпечення для анімаційної графіки. Воно використовується для створення 3D-анімації, ігор,

відстеження руху тощо.

Розробник: Blender Foundation

Ліцензія: Безкоштовне програмне забезпечення

Операційна система: FreeBSD, Linux, Mac OS X, Windows

Сайт розробника: <https://www.blender.org/>

- 13) **Maya** – програма для комп’ютерної 3D-графіки та анімації, яке особливо корисне для створення інтерактивних 3D-додатків, фільмів, відеоігор, телешоу тощо.

Розробник: Autodesk

Ліцензія: умовно-безкоштовне (пробний період)

Операційна система: Apple OS X, Microsoft Windows, Linux

Сайт розробника: <https://www.dpbolvw.net/click-9084201-12160174>

Онлайн-сервіси для створення комп’ютерної анімації:

- **Animaker** – онлайн-платформа для створення 2D-анімації, яка має велику кількість шаблонів та інструментів для створення анімації.

Посилання: <https://www.animaker.com/>

- **Canva Animator** – онлайн-платформа для створення анімаційних графічних елементів, яка має велику кількість готових ефектів та інструментів.

Посилання: https://www.canva.com/uk_ua/

- **Imgflip** – онлайн-сервіс для створення гіфок зі своїми зображеннями або з використанням готових шаблонів.

Посилання: <https://imgflip.com/>

Практичні завдання до уроків

Тема уроку: Практична робота №2

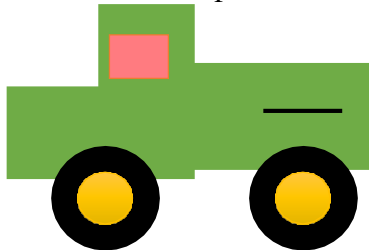
«Створення простого анімованого зображення»

Мета уроку: навчитись створювати прості анімаційні зображення у середовищі редактора Microsoft PowerPoint.

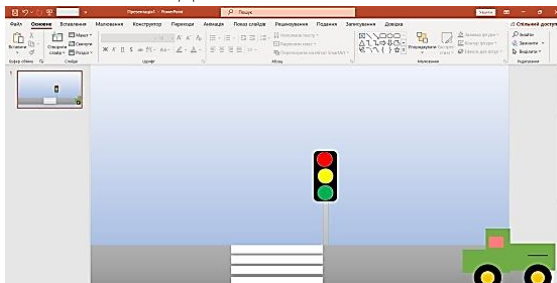
Завдання «Машинка»: Створити анімацію в PowerPoint зі створених об'єктів (автомобіль, світлофор, дорога) за зразком *Анімація в PowerPoint.mp4*, в якій машинка, залежно від кольору світлофору, рухається по дорозі.

Хід виконання завдання:

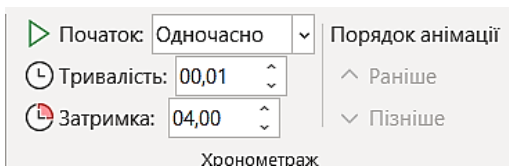
1. Відкрийте програму Microsoft PowerPoint. Створіть один слайд, оберіть макет: Пустий слайд.
2. Змініть колір фону слайда на **Блакитний** (*Конструктор =>Тло =>Стилі тла => Формат тла*).
3. Використовуючи інструмент **Фігури** (*Вставка => Фігури*) будуюмо автомобіль. Даний об'єкт може бути створений за Вашою власною фантазією.



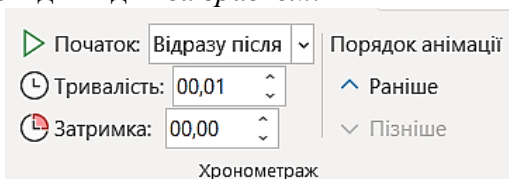
4. Використовуючи інструмент **Фігури** (*Вставка => Фігури*) Малюємо дорогу та світлофор. В результаті має вийти такий слайд:



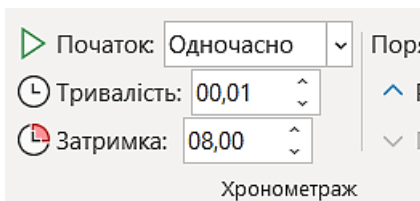
5. Додайте до фігури *Світлофор*, а саме *Жовте коло* анімацію **Зникнення** з групи *Вихід*.
6. У групі *Хронометраж*, змініть властивість відповідно за зразком:



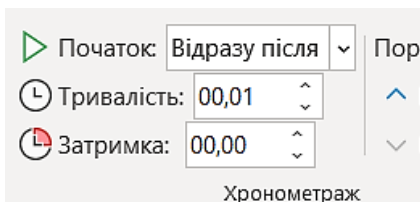
7. Додайте до *Червоного кружечку* анімацію **Звичайна поява** з групи *Вхід*.
8. У групі *Хронометраж*, змініть властивості **Червоного кружечка** відповідно за зразком:



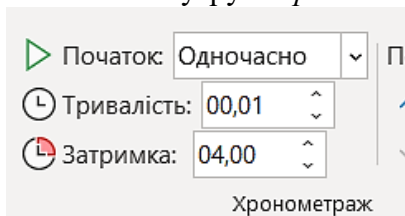
9. До *Червоного кружечку* додаємо ще одну анімацію **Зникнення** з групи *Вихід*. У групі *Хронометраж*, змініть властивості **Червоного кружечка** відповідно до зразка:



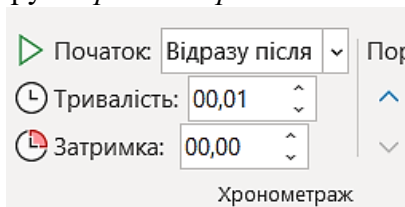
10. Далі додаємо анімацію **Звичайна поява** з групи *Вхід* до об'єкту *Жовтого кружечку*, аналогічно на цій анімації у групі *Хронометраж*, змінюємо властивості відповідно до зразка:



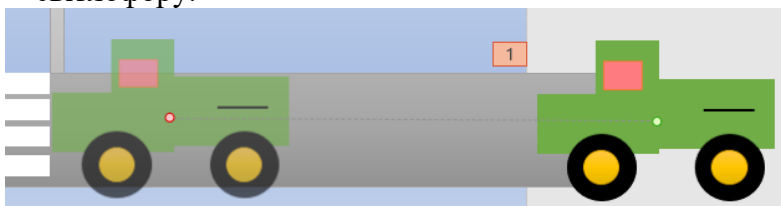
11. Після цього **Жовтий колір має зникну**, тому додаємо ще одну **анімацію Зникнення** з групи **Вихід** та налаштуємо властивості у групі **Хронометраж**:




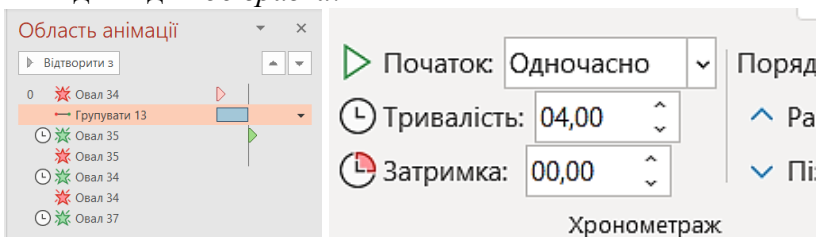
12. До об'єкта **Зелений кружечок** додаємо **анімацію Звичайна поява** з групи **Вхід** та налаштуємо властивості у групі **Хронометраж**:



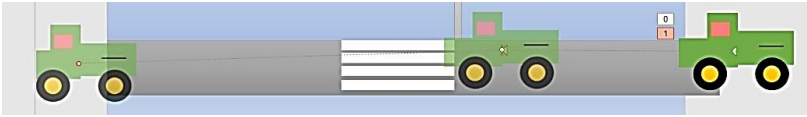
13. До об'єкта **Машина** додаємо **анімацію Ліворуч** з групи **Інші шляхи переміщення** та розташуйте **траєкторію руху** автівки **максимально близько** **перетягуємо** до **світлофору**:



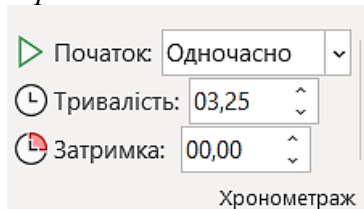
14. Налаштуємо властивості у групі **Хронометраж** та **пересунемо** **анімацію вгору** (використовуючи ) **відповідно до зразка**:



15. До об'єкта **Машина** додаємо ще одну **анімацію Ліворуч** з групи **Інші шляхи переміщення**. Далі розміщуємо так: де **зелена точка** – ближче до світлофору, а **червона** – за поле:

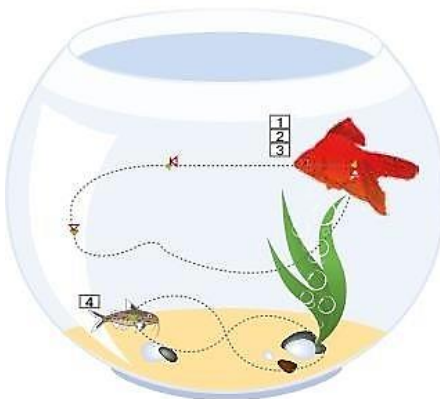


16. Налаштовуємо властивості у групі **Хронометраж** відповідно до зразка:



17. Перегляньте презентацію в режимі демонстрації.

Завдання «Акваріум»: Створити однослайдову презентацію із зображенням акваріума та зімітуйте рух рибок у ньому за траскторіями.



Дотримуйся таких вимог:

- Рибки мають рухатися постійно, до кінця показу слайда.
- Рух рибок має бути рівномірним, без прискорень та уповільнень.

- Золота рибка має пропливати повне коло за 10 с, а сомик – за 5 с.
- Рибки не повинні рухатися задом наперед, тобто вони мають повертатися внапрямок руху. Для цього одночасно з ефектом руху до рибок має застосовуватися один або кілька ефектів обертання (з категорії Виокремлення). Налаштуй їхні параметри так, щоб рух рибок виглядав якомога реалістичніше.

Примітка: Необхідні фото знайдіть у мережі Інтернет.

Тема уроку: Практична робота №3 «Створення GIF-анімацій».

Мета уроку: навчитися створювати та редагувати файли, які містять GIF – анімацію із застосуванням програми GIMP, налаштовувати часові параметри, правила компонування об'єктів, імпортувати дані із зовнішніх джерел.

Завдання «Метелик»: Створіть GIF-анімацію руху метелика, який літає на квітковому полі у програмі GIMP.

Примітка: необхідні для виконання завдання матеріали знаходяться у папці «Створення GIF-анімацій».

Інформатики > Створення GIF-анімацій



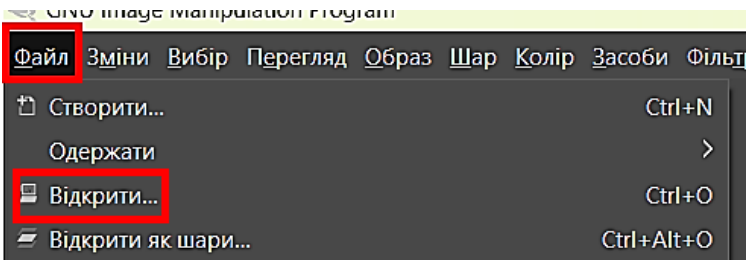
Квіткове
поле.jpg



Метелик.p
ng

Хід виконання завдання:

1. Відкрийте програму GIMP. Завантажте файл «Квіткове поле», який буде фоном для анімації.

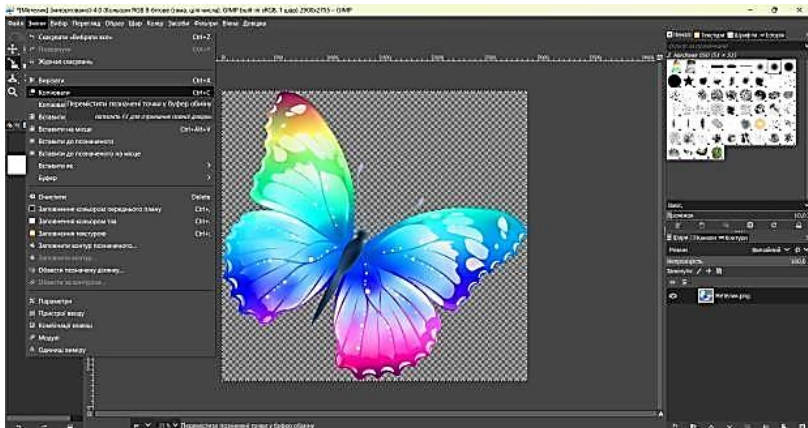


2. Відкрийте файл «Метелик» та скопіюйте зображення метелика на Квіткове поле, для цього:

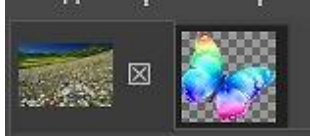
- **Виділіть** повністю шар(CTRL+A). Після цього у Вас буде видно осьтаку бігучу штрих пунктирну лінію:



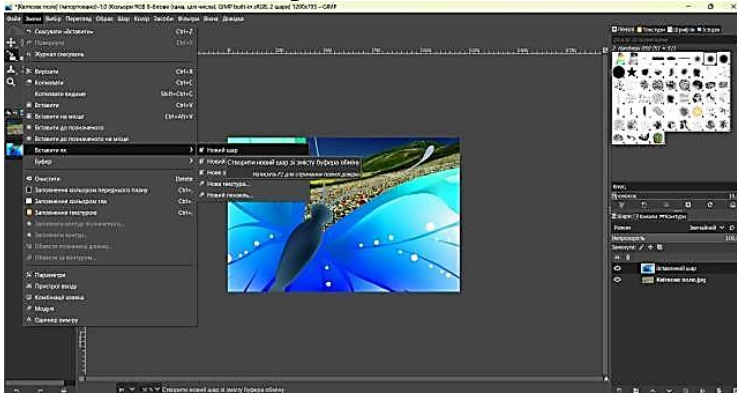
- Перейдіть на вкладку «Зміни» та оберіть команду **Копіювати**:



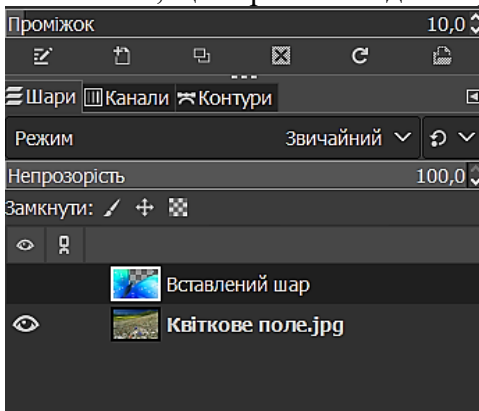
- **Перейдіть до фону**, натиснувши на піктограму вгорі:



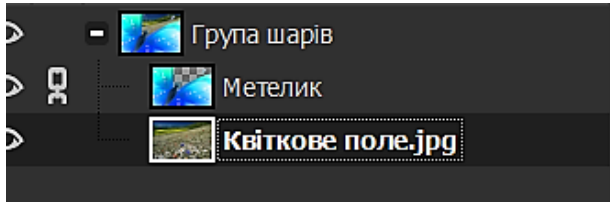
- Перейдіть на вкладку «Зміни» та оберіть команду **Вставити як шар**:



- Зліва ми бачимо, що отрималось два шари:



3. Створіть групу шарів ;
4. **Перемістіть** туди **вставлений фон** (у прикладі його перейменовано на «метелик») та **квіткове поле**:

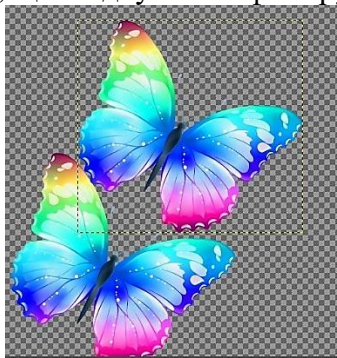


5. Далі ми з Вами ці групи шарів будемо дублювати. І на кожному з них щось змінювати, для цього:


- Натисніть кнопку  та зробіть штук 5 копій:



- Тепер метелика із шару **Копія Кадр 1**, трохи зміщуємо, щоб відбувався ефект руху:



6. Тепер на шарі **Копія Кадр 1** зробимо **трансформацію метелика**, він має не лише переміщуватися, але й складати крильця, **тому** обираємо інструмент **Об'єднане**

перетворення .

7. Обираємо метелика, якого будемо деформувати та трохи

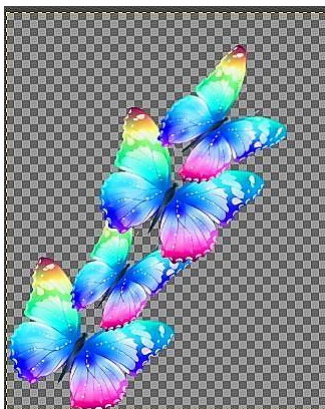
стискаємо, ніби він махає крильцями та натискаємо Enter:



8. Переходило до наступної групи шарів **Копія Копія Кадр 1** та робимо метелику ефект, ніби він летить:



9. Аналогічно пророблюємо з іншими метеликами в групах шарів:



10.Робимо видимим фон:



11.Перейдіть до вкладки «Фільтр» і виберіть «Анімація» => «Відтворення».



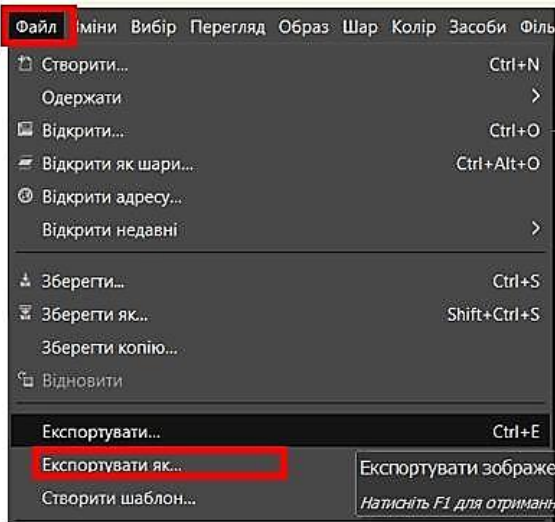
12.У вікні «Анімація» Ви можете змінити параметри анімації, такі як *розмір, швидкість та інші.*

13.Проекспериментуйте з часом відображення кожного кадру, щобстворити більш плавну анімацію.

14.Зберігаємо проєкт:



15. Зберігаємо анімацію в форматі gif.

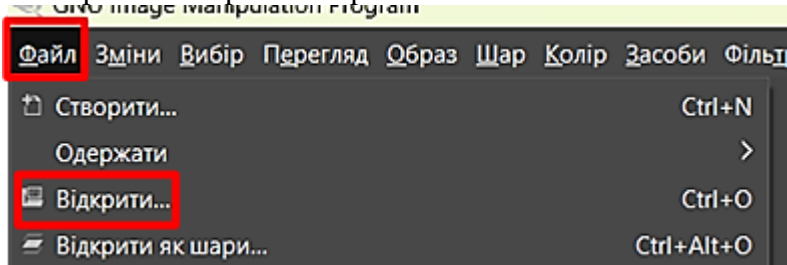


Завдання «Мій настрої»: Сфотографуй себе в різних емоціях (веселий, сумний, зацікавлений, серйозний та ін.) або знайдіть фото у мережі Інтернеті і створи GIF-анімацію «Мій настрої» у програмі GIMP.

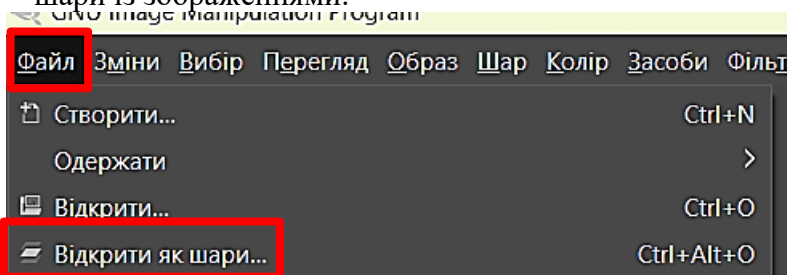
Хід виконання завдання:

1. Завантажте фотографії Ваших емоцій, які ви хочете анімувати, у GIMP, вибравши «Файл» => «Відкрити» та

вибравши потрібне зображення.



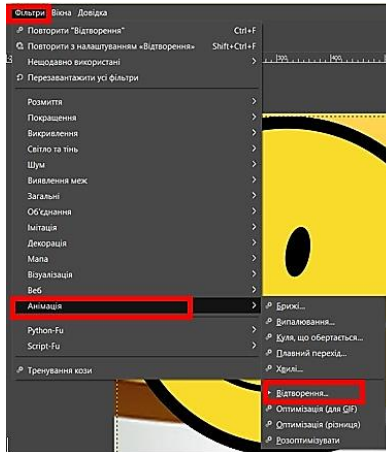
2. **Створіть новий шар** для другої емоції, наприклад, «радість». Для цього в меню «Файл» оберіть «Відкрити як шар» і виберіть зображення з вашого комп'ютера. Повторіть цей крок для кожної емоції, додаючи нові шари із зображеннями.



3. **Розмістіть шари** в порядку відповідного кадру анімації, який ви визначили раніше. *Наприклад*, якщо ви хочете, щоб на першому кадрі з'явилася фото з емоцією «радість», на другому – емоція «сум», а на третьому – емоція «зацікавленість», то перший шар повинен надрукувати зображення з емоцією «радість», другий – емоцію «сум», а третій – емоція «зацікавленість».



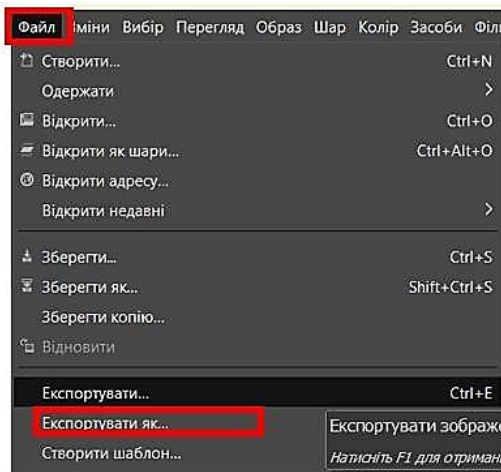
4. Після того, як створили всі необхідні шари з Вашими фотографіями, виберіть усі шари за допомогою натискання клавіші «Ctrl» і виділіть назви кожного шару.
5. Перейдіть до вкладки «Фільтр» і виберіть «Анімація» => «Відтворення».



6. У вікні «Анімація» Ви можете змінити параметри анімації, такі як *розмір, швидкість та інші*. Виберіть кожний шар відповідно до потрібної емоції та вкажіть час, протягом якого шар буде відображатися.



7. Після налаштування анімації натисніть *кнопку «Відтворити»* для перегляду створеної анімації. Якщо все вірно, *натисніть «Файл» =>«Експорт»* для збереження GIF-анімації.

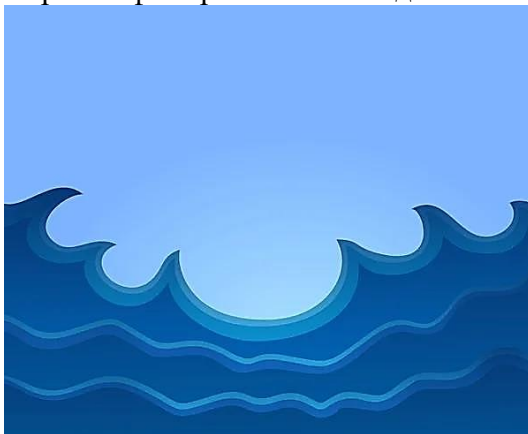


Примітка: Щоб отримати кращі результати, використовуйте

інші інструменти, наприклад різні типи пензлів, фільтри та ефекти. Також варто експериментувати з часом відображення кожного кадру, щоб створити більш плавну анімацію.

Завдання «Рух води»: Використовуючи графічний редактор GIMP, підготувати *серію послідовних зображень розливання води*, де хвилі та розбризкування показують рух води, та **створити GIF-анімацію «Рух води»** у програмі GIMP.

Примітка: Використовуйте пензель з плавною м'якістю, щоб створити ефект розливання води.



Тема уроку: *Анімація тексту. Практична робота «Анімація тексту».*

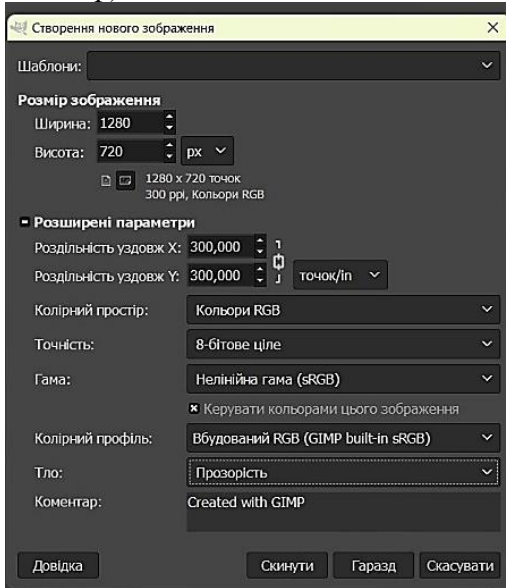
Мета уроку: ознайомити учнів із анімацією тексту та застосувати на практиці навички створювати анімований текст в растровому графічному редакторі GIMP

Завдання «Моє прізвище»: Створіть анімований текст Вашого прізвища та ім'я, де кожен символ поступово з'являється на екрані у програмі GIMP.

Хід виконання завдання:

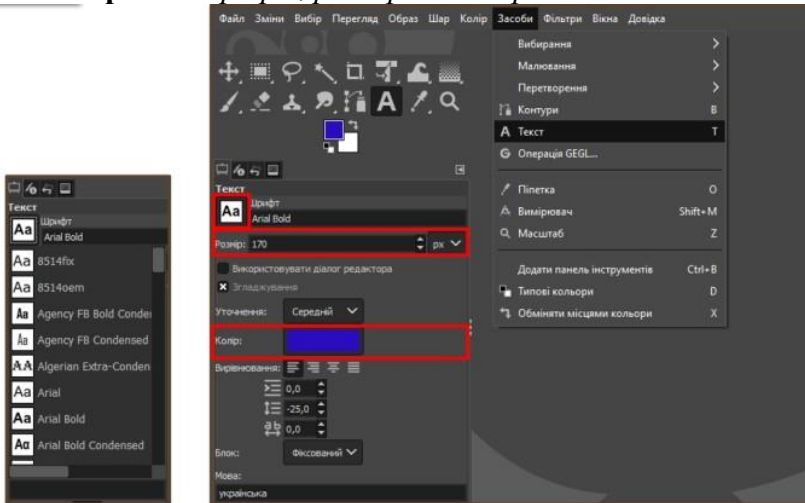
1. Відкрийте програму GIMP. Створіть новий документ (*Файл => Створити => Налаштовуємо*

розмір та колір)

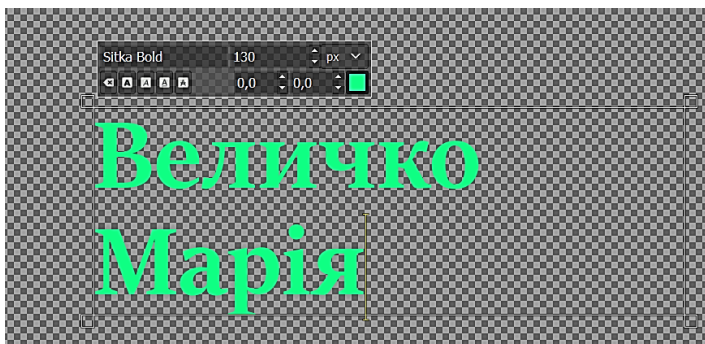



2. На панелі інструментів обираємо **Текст** .

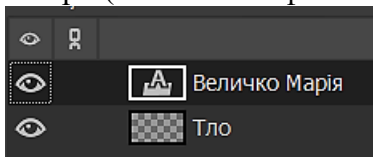
3. Обираємо шрифт, розмір та колір за бажанням.





4. Обираємо область та друкуємо текст:

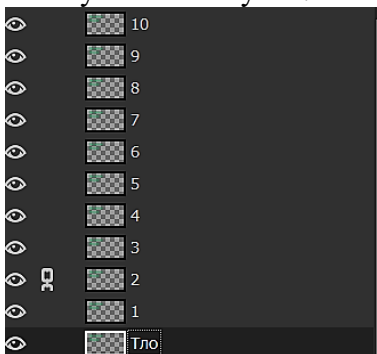


5. Обираємо інструмент Переміщення  та розміщуємо тест вбудь – якому зручному місці;
6. Об'єднує два шари (тестовий та фоновий)




для цього: натискаємо праву кнопку миші та обираємо команду  **Об'єднати з попереднім**

7. Для того, щоб створити анімацію, необхідно створити стільки шарів, скільки літер в тексті, тому копіюємо шари, використовуючи кнопку .




8. Починати створювати анімацію будемо з самого нижнього шару, всі інші вимикаємо:



9. Тепер обираємо інструмент Гумка , обираємо товщину і повністю прибираємо текст:



10. Відкриваємо наступний шар, обираємо інструмент Гумка  та прибираємо все, ОКРІМ першої літери:

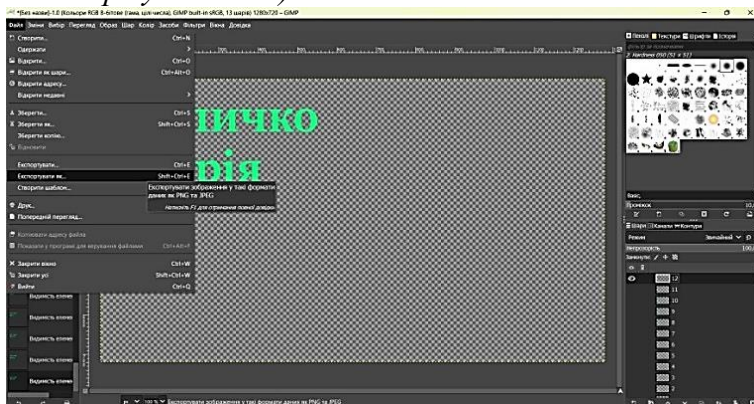


11. Відкриваємо наступний шар, обираємо інструмент Гумка та прибираємо все, ОКРІМ перших *двох літер*:

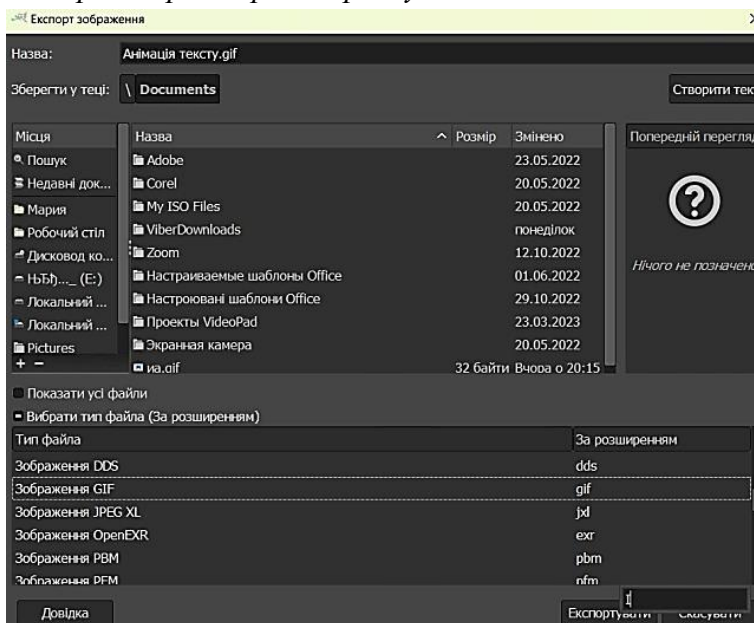


12. Аналогічно пророблюємо з іншими шарами, доки не дійдемо до останнього, де буде відобразитися ВЕСЬ текст.

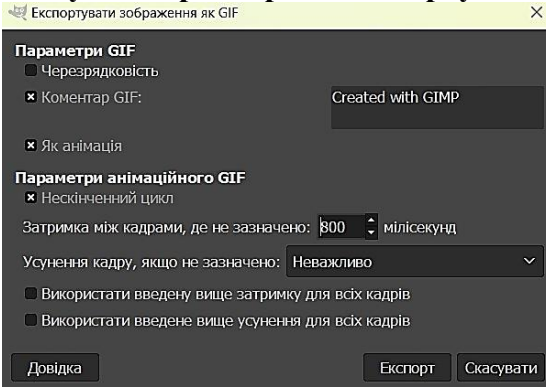
13. Тепер необхідно **зберегти** анімацію (*Файл => Експортувати як..*)



14. *Обираємо розширення файлу:*



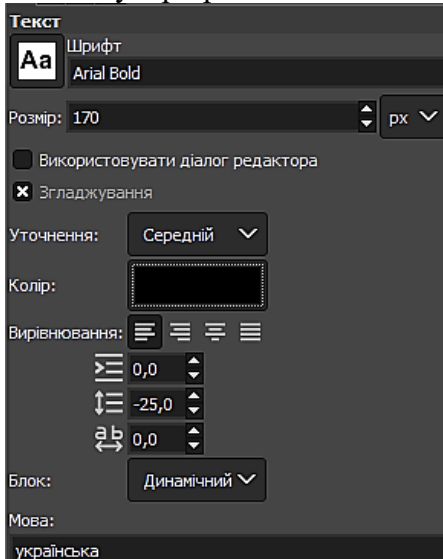
15. Налаштуємо параметри та експортуємо анімацію:

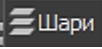
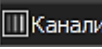
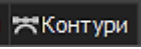



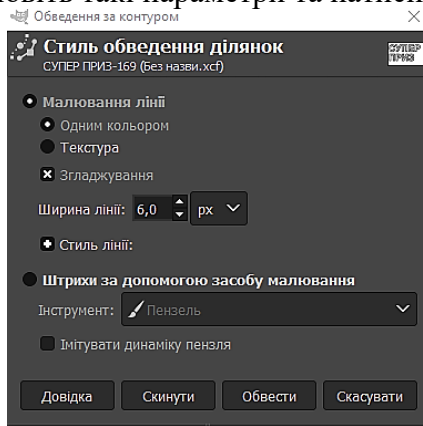
Завдання 2. Переглянути анімацію **Задання 2.gif**.
Створити анімацію тексту з літерами, що переливаються у програмі GIMP.


Хід виконання завдання:

1. **Відкрийте** середовище GIMP (*Пуск => Усі програми => GIMP*).
2. **Створити нове зображення** розміром 640x400 пікселів білого кольору.
3. **Вставте текст** «Супер приз» із такими властивостями:



4. **Обведіть літери.** Для цього клікніть ПКМ по назві шару і оберіть команду **Текст та контур**. Створіть новий прозорий шар такого ж розміру та назвіть його **Обведення**.
5. Змініть колір тексту на синій.
6. Перейдіть в меню **Контури**    та натисніть на кнопку **Обвести контур вказаним чином** . Встановіть такі параметри та натисніть **Обвести**.



7. Розмийте контур тексту **Фільтр** => **Розмиття** => **Розмиття за Гаусом** (параметри за замовчуванням)
8. Створіть новий файл з такими ж параметрами, як і попередній.
9. За допомогою інструменту Градієнт створіть веселкове тло (Інструмент **Градієнт** => Обрати веселкову заливку  **Full saturation spe** => Провести вертикальну лінію від гори до низу).
10. Скопіюйте шар **Тло**.
11. Збільшіть розмір полотна (**Образ** => **Розмір полотна** => **Висота 800**).
12. Зсуньте копію шару Тло до низу, таким чином, щоб низ полотна співпадав із низом шару. (Використайте інструмент **Переміщення**)
13. **Об'єднайте** ці два шари.

14. Скопіюйте цей файл у файл із надписом. (меню **Вибір** => команда **Всього** => меню **Зміни** => команда **Копіювати** => перейти у файл із написом => меню **Зміни** => команда **Вставити**)
15. У вікні шарів натисніть ПКМ на плаваючий шар і оберіть команду **Уновий шар**.
16. Закрийте без збереження файл із веселковим тлом.
17. Здвиньте веселкове тло вгору таким чином, щоб його низ співпав із низом зображення.
18. Для зручності перейменуйте шар. Наприклад *Веселка*.
19. Зробіть копію шару *Веселка* і зсуньте її трохи вниз.
20. Зробіть копію активного шару та також зсуньте її вниз, повторіть цідії до тих пір, поки верх останньої копії не співпаде із верхом інших шарів (має бути приблизно 7 копій).
21. Активуйте шар із текстом. Клікніть праву кнопку мишки на назві та оберіть команду **Альфа-канал** => **Позначене**. Виділились літери.
22. Зробіть інверсію виділення. (**Вибір** => **Інвертувати**)
23. Активуйте шар *Веселка* та натисніть клавішу DEL.
24. Перейдіть на копію шару *Веселка* і знову натисніть DEL. Повторіть цідії із кожним веселковим шаром. Таким чином досягнете ефекту, що веселкові будуть лише літери, а тло видалиться.
25. За допомогою інструменту **Кадрування** видалить все зайве.
26. Активуйте шар *Обведення* та створіть його копію.
27. Перемістіть копію шару *Обведення* над шаром *Веселка*. Зробіть ще одну копію шару *Обведення* та знову перемістіть його над шаром із веселковими літерами, потрібно зробити стільки копій шару *Обведення* скільки шарів із веселковими літерами і розташувати їх таким чином, щоб шар *Обведення* був над шаром із веселковими літерами. Верхнім має бути шар із веселковими літерами.
28. **Об'єднайте шари попарно** (шар із *Обведенням* та шар




- із веселковими літерами), вийшло приблизно 8 об'єднаних шарів.
29. Шари із текстом та тлом **видаліть**.
 30. Перегляньте анімацію, що вийшла.
 31. Збережіть проєкт у свою папку із назвою *Завдання 2*.
 32. Експортуйте в анімацію (**Файл => Експортувати як => Обрати типЗображення GIF**).
 33. Закрийте редактор GIMP. Відкрийте файл Завдання 2.gif за допомогою браузера чи програми для перегляду відео.

Тема уроку: *Практична робота «Створення векторної анімації».*

Мета уроку: навчити створювати просту анімацію у векторноуграфічному редакторі Synfig Studio.

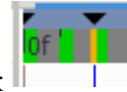
Завдання 1: Створити анімацію руху об'єкта.

Хід виконання завдання:

1. Відкрийте програму Synfig Studio (**Пуск =>Усі програми => Synfig Studio**).
2. Оберіть інструмент Коло на панелі інструментів. Встановіть заливку фігури на свій розсуд. Намалюйте коло довільного радіусу. Перемістіть коло у лівий нижній куточок робочої області. Для цього активуйте інструмент Трансформації та перетягніть коло за зелену точку, що знаходиться в центрі.
3. Ввімкніть режим анімації у правому нижньому кутку Робочої області . Після натиснення кнопка виглядатиме так .
4. Створіть перший ключовий кадр, для цього на Шкалі часу встановіть синю вертикальну рису на 6f , на блакитній частині натисніть ПКМ та оберіть команду

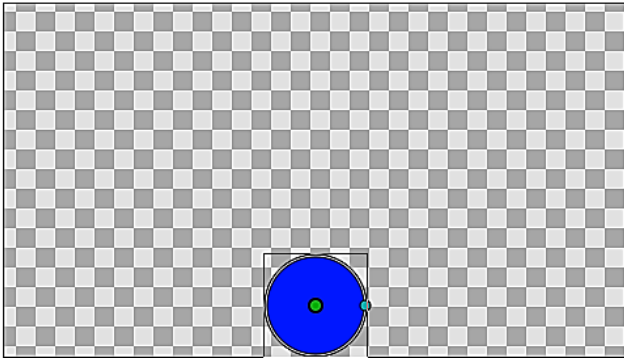


, в результаті з'явиться

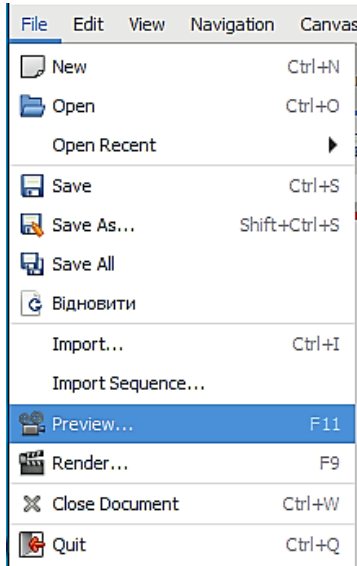


на Панелі ключових кадрів чорний трикутник

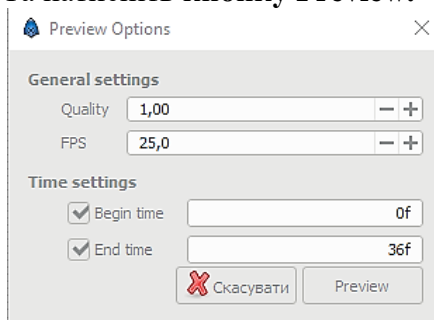
5. Перемістіть коло у нижню центральну точку робочої області.




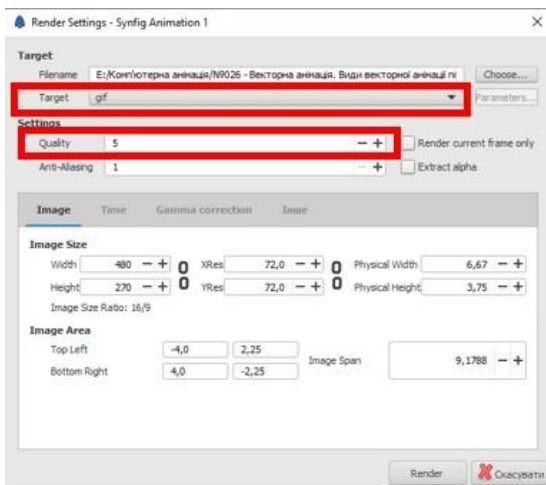
6. Створіть другий ключовий кадр 12f.
7. Перемістіть коло у правий нижній куточок робочої області.
8. Створіть третій ключовий кадр 18f.
9. Перемістіть коло у правий верхній куточок робочої області.
10. Знову створіть ключовий кадр 24f та перемістіть коло у верхню центральну точку робочої області.
11. Додайте ще один ключовий кадр 30f та перемістіть коло у лівий верхній куточок робочої області.
12. І останній ключовий кадр 36f, перемістіть коло у лівий нижній куточок.
13. Перегляньте анімацію, що вийшла. Для цього оберіть у меню *File* команду **Preview**.



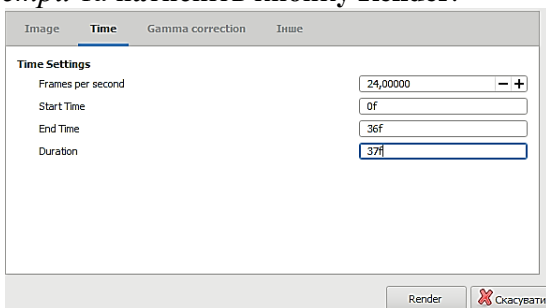
14. У вікні що відкриється *встановіть наступні параметри* (параметр End time має відповідати кінцевому кадру анімації). Та натисніть **кнопку Preview**.



15. У вікні, що відкрилось натисніть кнопку відтворити 
16. *Збережіть проєкт* у власну папку із назвою Анімація руху. (*File => Save as ... => Вказати шлях => Назва => Зберегти*)
17. Створіть **gif**. Для цього оберіть у меню файл **команду Render**, у вікні що відкриється встановіть наступні параметри



18. Перейдіть у вкладку **Time** і встановіть наступні параметри та натисніть кнопку **Render**:



Завдання 2: Створіть анімацію в якій коло перетворюється на серце, змінює колір та розмір.

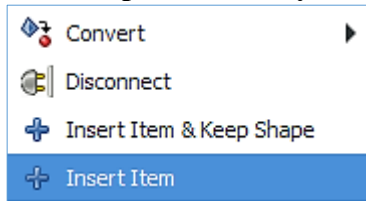
Хід виконання завдання:

1. Відкрийте програму Synfig Studio (*Пуск => Усі програми => SynfigStudio*).
2. Намалуйте коло для трансформації колір та розмір на ваш розсуд. *Зверніть увагу*, коли активуєте інструмент Коло необхідно в налаштуваннях цієї фігури Create a region layer та Create an outline layer

Layer Type:






3. Розташуйте по центру робочої області.
4. Ввімкніть режим анімації.
5. Створіть ключовий кадр 48f.
6. За допомогою управляючих точок **перетворіть коло на серце** (або іншу фігуру на ваш розсуд). Для того щоб додати управляючі точки вмісті де має бути точка клікніть ПКМ та **оберіть команду Insert Item**



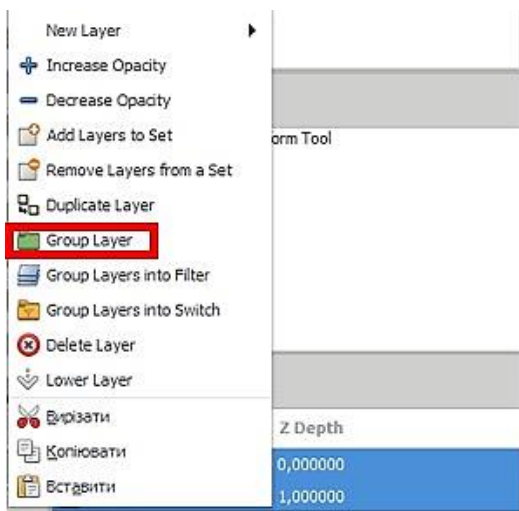
Для зручності можна збільшити зображення натиснувши клавішу Ctrl та прокрутивши колесо мишки.

7. Після того як досягли бажаної форми **змініть колір**.

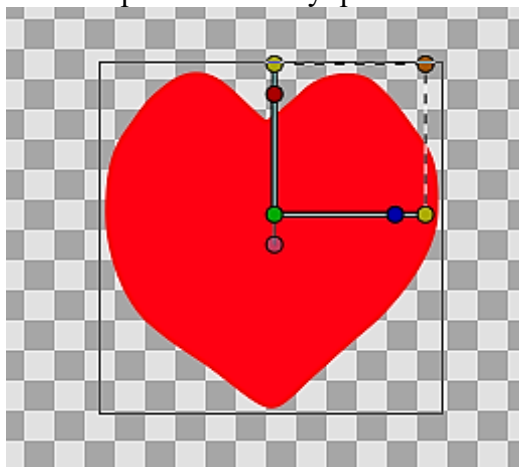
Для цього активуйте інструмент заливка  оберіть бажаний колір та натисніть на фігуру. Зверніть увагу, що контур фігури може залишитись іншого кольору його теж потрібно замінити. У вікні властивості фігури оберіть відповідний колір

Name	Value
π Z Depth	0,000000
π Opacity	1,000000
Blend Method	Composite
Konip	
Origin 	-1,005827px,0,614674px
Invert	<input type="checkbox"/>
Antialiasing	<input checked="" type="checkbox"/>
π Feather	0,000000px

8. Створіть новий ключовий кадр 96f.
9. Збільшіть розмір отриманої фігури. Для цього потрібно виділити обидва шари у вікні Шари клікнути ПКМ і створити групу шарів



10. В результаті отримаємо такі управляючі точки.



11. Потягніть за верхню праву точку по діагоналі і збільшіть фігуру добажаних розмірів.
12. **Перегляньте анімацію**, що вийшла. Для цього оберіть у меню File команду Preview.
13. Збережіть проект у свою папку із назвою **Зміна форми**. (Зберегти як=> Вказати шлях до своєї папки)
14. Створіть gif. У вкладці Time встановіть кінець анімації 96f.

Підсумковий тест з вибіркового модуля «Комп'ютерна анімація»

Завдання з однією правильною відповіддю¹

1. Комп'ютерна анімація – це....
 - а) відео, яке переглядається за допомогою комп'ютера;
 - б) вид анімації, в якому об'єкти виконуються за допомогою комп'ютерних засобів;
 - в) вид анімації, в якому об'єкти створюються за допомогою комп'ютерних засобів;**
 - г) анімація, яку складає комп'ютер;
2. Що необхідно мати для створення анімації?
 - а) сценарій;**
 - б) художні навички;
 - в) аркуш формату А4;
 - г) вчителя інформатики;
3. Шар в комп'ютерній анімації – це.....
 - а) накладення зображень;
 - б) однорідна за якоюсь ознакою частина чогось;
 - в) аркуш;
 - г) елемент, який використовується для створення зображень або тексту;**
4. Що є основним інструментом при створенні покадрової анімації?
 - а) звукова шкала;
 - б) відеошкала;
 - в) часова шкала;**
 - г) музична шкала;
5. Анімація – це....
 - а) намальовані від руки малюнки;
 - б) рух намальованих зображень;**
 - в) фрагмент відео;
 - г) швидкісне відео;

¹ Правильні відповіді виділено жирним шрифтом.

Завдання на відповідність

6. Установіть відповідність між видом анімації(1-6) та його зображенням(А-Д):

1. Лялькова анімація	А) 
2. Пластилінова анімація	Б) 
3. Тіньова анімація	В) 
4. Мальована класична анімація	Г) 
5. Сипка анімація	Д) 
6. Перекладна анімація	

Відповідь: 1 – Г; 2. – А; 3 – Д; 4 – Б ; 5 – В;

7. Установіть відповідність між назвою програми(1-5) для анімації та її піктограмою(А-Д):

1. GIMP	
2. Synfig Studio	
3. Adobe Photoshop	
4. Pencil2D	
5. Adobe Animate	

Відповідь: 1 – Г; 2 – Б; 3 – Д; 4 – В; 5 – А.

Запитання з декількома правильними відповідями:

8. Які існують **види** комп'ютерної анімації:

а) 3-d анімація;

б) 2-d анімація;

в) 5-d анімація;

9. Оберіть формати файлів **векторних зображень**:

а) BMP

б) WMF;

- в) CDR ;**
- г) PNG;

10. Для розміщення на веб-сторінках в Інтернеті використовують **растрові зображення** у форматах:

- а) HTML;
- б) GIF;**
- в) EXE;
- г) JPG;**
- д) BMP;**

11. Вкажіть особливості растрової GIF – анімації:

- а) Можливість задавання величини затримки кожного кадру на екрані;**
- б) Можливість задавання кількості повторень або зациклювання послідовності кадрів;**
- в) Не потрібно промальовувати кожен кадр, а тільки ключові і задати типпереходу між ними;
- г) Кожен наступний доданий до анімації кадр збільшує розмір файлу;**
- д) Невеликий інформаційний об'єм файлів;

12. Вкажіть **особливості векторної анімації**:

- а) Можливість задання величини затримки кожного кадру на екрані;
- б) Не потрібно промальовувати кожен кадр, а тільки ключові і задати типпереходу між ними;**
- в) Можливість задання кількості кадрів за секунду;**
- г) Кожен наступний доданий до анімації кадр збільшує розмір файлу.
- д) Невеликий інформаційний об'єм файлів.**

Вибірковий модуль «Інформаційна безпека» (17 годин)

Укладач С.Д.Настусенко

Календарно-тематичний план

Посилання:

<https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-10-11-klas/2018-2019/informatika-standart-10-11.docx>

Номер уроку	Тема	Зміст навчального матеріалу
1/1	Тема 1. Основи безпеки інформаційних технологій	Основні поняття в області безпеки інформаційних технологій.
2/2		Основні причини загострення проблеми забезпечення безпеки інформаційних технологій. Інформація та інформаційні відносини.
3/3		Загрози безпеці інформації в автоматизованих системах.
4/4		Технічні та програмні засоби добування інформації.
5/5		<i>Практична робота №1:</i> «Використання програмних засобів для тестування та очищення ОС від вірусів та шкідливого ПЗ»
6/1	Тема 2. Забезпечення безпеки інформаційних технологій	Об'єкти захисту. Види заходів протидії загрозам безпеки.
7/2		Правові основи забезпечення безпеки інформаційних технологій.
8/3		Ідентифікація та аутентифікація користувачів. Розмежування доступу до ресурсів автоматизованих систем.
9/4		Контроль цілісності програмних і інформаційних ресурсів. Виявлення атак.
10/5		<i>Практична робота №2:</i> «Використання засобів адміністрування ОС для налаштування прав користувачів щодо доступу до інформаційних ресурсів»

Номер уроку	Тема	Зміст навчального матеріалу
11/1	Тема 3. Забезпечення безпеки комп'ютерних систем і мереж	Проблеми забезпечення безпеки в комп'ютерних системах і мережах. Засоби захисту мереж.
12/2		Призначення, можливості, і основні захисні механізми міжмережевих екранів (брандмауерів).
13/3		Основні захисні механізми. Політика безпеки при доступі до мережі загального користування.
14/4		Системи аналізу вмісту поштового і веб-трафіку. Віртуальні приватні мережі (VPN).
15/5		Практична робота «Створення віртуальних приватних мереж»
16/6		Практична робота «Налаштування засобів системного міжмережевого екрану (брандмауера).
17/7		Антивірусні засоби захисту.
Разом: 17 годин		

Список рекомендованої літератури для вчителя

1. Глинський Я.М. Інформатика: 10-11 класи: у 2-х частинах. Ч.2 інформаційні технології. 5-те видання. Львів: Деол, СПД Глинський, 2005. 264 с.
2. Інформаційна безпека інформаційно-комунікаційних систем. Лабораторний практикум. Частина 1. – Комплекси засобів захисту інформації від НСД: навч. посібник / Д.В. Голев, В.Й. Кільдишев, В.Г. Кононович ; за ред. чл.- кор. МАЗ В.Г. Кононовича. – Одеса: ОНАЗ ім. О.С. Попова, 2010. – С.176. - Режим доступу: <https://app.box.com/s/ptc5dzuqfo8go2o1pckxogpye9xr4tzx>
3. Інформаційна безпека інформаційно-комунікаційних систем. Лабораторний практикум. Частина 2. – Комплекси технічного захисту інформації Навч. посібник / Д. В. Голев, О.Ю.Русляченко, Ю.В.Белова,

- Д.С.Гончарук ; за ред. чл.-кор. МАЗ В.Г. Кононовича.– Одеса: ОНАЗ ім. О.С. Попова, 2010. – С. 184. – Режим доступу: <https://app.box.com/s/sb7ggqzrtrh2bgl2nprn6b74ma5d849i>
4. Єсін В. І. Безпека інформаційних систем і технологій : навчальний посібник / В. І. Єсін, О. О. Кузнецов, Л. С. Сорока. Х. : ХНУ імені В. Н. Каразіна, 2013. 632 с.
 5. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закл. / Коршунова О.В., Завадський І.О., Стасюк З.Р., Потієнко В.О. Київ : Видавничий дім «Освіта», 2022. 257 с.
 6. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закл. / [О.О. Бондаренко, В.В. Ластовецький, О.П. Пилипчук, Є.А. Шестоपालов]. Харків : Вид-во «Ранок», 2022. 237 с.
 7. Інформатика : підруч. для 9 класу з поглибленим вивченням інформатики загальноосвіт. навч. закл. / Руденко В.Д., Речич Н.В., Потієнко В.О. Харків : Вид-во «Ранок», 2022. 280 с.
 8. Інформаційна безпека в комп'ютерних мережах : навч. посіб. / О. А. Смірнов, О. К. Коноплицька-Слободенюк, С. А. Смірнов [та ін.] ; М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т. Кропивницький : Лисенко В.Ф., 2020. 295 с.
 9. Інформаційна безпека держави: навч. посіб. для студ. спец. 6.170103 «Управління інформаційною безпекою», 125 «Кібербезпека»/ В.І. Гур'єв, Д.Б. Мехед, Ю.М. Ткач, І.В. Фірсова. Ніжин: ФОП Лук'яненко В.В. ТПК «Орхідея», 2018. 166 с.
 10. Кавун С. В. Інформаційна безпека. Навчальний посібник / С. В. Кавун, В. В. Носов, О. В. Манжай. Харків: Вид. ХНЕУ, 2008. 352 с. Режим доступу: <http://www.repository.hneu.edu.ua/jspui/handle/123456789/3068>
 11. Інформаційна безпека. Підручник / В. В. Остроухов, М. М. Присяжнюк, О.І. Фармагей, М. М. Чеховська та ін.;

під ред. В. В. Остроухова. К.:Видавництво Ліра-К, 2021. 412 с.

12. Інформатика : 10 клас: підруч для загальноосвіт.навч. закл. : рівень стандарту / Й.Я. Риквінд, Т.І. Лисенко Л.А.Чернікова, В.В.Шакотько; за заг. ред. М.З. Згуровського. Київ. : Генеза, 2010. 296 с.
13. Хорошко В.О. Основи інформаційної безпеки /Хорошко В.О., Чередниченко В.С., Шелест М.Є./ За ред. проф. В.О. Хорошка. К.: ДУІКТ, 2008.

Інтернет джерела:

1. https://courses.prometheus.org.ua/courses/course-v1:Internews+INFOS101+UA_2021_T3/about
2. https://courses.prometheus.org.ua/courses/KPI/IS101/2014_T1/about
3. <https://skillsforall.com/course/introduction-to-cybersecurity?userLang=uk-UA>
4. <https://zillya.ua/prometheus>

Список рекомендованої літератури для учнів

1. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закл. / Коршунова О.В., Завадський І.О., Стасюк З.Р., Потієнко В.О. Київ : Видавничий дім «Освіта», 2022. 257 с.
2. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закл. / [О.О. Бондаренко, В.В. Ластовецький, О.П. Пилипчук, Є.А. Шестопапов]. Харків : Вид-во «Ранок», 2022. 237 с.
3. Інформатика : підруч. для 9 класу з поглибленим вивченням інформатики загальноосвіт. навч. закл. / Руденко В.Д., Речич Н.В., Потієнко В.О. Харків : Вид-во «Ранок», 2022. 280 с.
4. Кавун С. В. Інформаційна безпека. Навч. посібник / С. В. Кавун, В. В. Носов, О. В. Манжай. Харків: Вид. ХНЕУ, 2008. 352 с. Режим доступу:

<http://www.repository.hneu.edu.ua/jspui/handle/123456789/3068>

5. Інформатика : 10 клас: підруч для загальноосвіт. навч. закл. : рівень стандарту / Й.Я. Риквінд, Т.І. Лисенко Л.А.Чернікова, В.В.Шакотько; за заг. ред. М.З. Згуровського. Київ. : Генеза, 2010. 296 с.

Інтернет джерела:

1. https://courses.prometheus.org.ua/courses/course-v1:Internews+INFOS101+UA_2021_T3/about
2. <https://skillsforall.com/course/introduction-to-cybersecurity?userLang=uk-UA>
3. <https://zillya.ua/prometheus>

Програмне забезпечення:

Антивіруси

Avast Free Antivirus – безкоштовна програма антивірусного захисту, яка надає базовий рівень захисту для комп'ютерної системи, включаючи захист в режимі реального часу, перевірку на віруси та інші загрози, а також функції оптимізації швидкості та продуктивності комп'ютера; має інтуїтивно зрозумілий інтерфейс та легка в установці та використанні; є можливість оновлювати базу даних вірусів та інших загроз в режимі реального часу, що допомагає захистити комп'ютер від нових загроз.

Розробник: AVAST (Чехія)

Ліцензія: Безкоштовно (персональне використання)

Система: Windows 7/8/10

Сайт: https://www.avast.ua/lp-ppc-hp-v5?ppc_code=012&ppc=a&gclid=Cj0KCQjww4-hBhCtARIsAC9gR3bxhUrHV-ZJeLtfWtkXyryMHA3sgSvE9y9HCevLp7uVQaDjaAaLlzMaAhCYEALw_wcB&gclsrc=aw.ds#pc

Avira Free Antivirus – безкоштовна антивірусна програма для захисту комп'ютера від шкідливих програм,

вірусів та інших загроз. Вона забезпечує базовий захист від вірусів, шпигунського програмного забезпечення, троянів та інших загроз, що можуть нашкодити комп'ютеру. Програма має простий інтерфейс та легко використовується навіть користувачами з мінімальним досвідом використання антивірусних програм. Окрім цього, Avira Free Antivirus має можливість сканування комп'ютера на наявність шкідливих програм, а також контролює вхідні та вихідні з'єднання з Інтернетом для захисту приватності.

Розробник: Avira GmbH (Німеччина)

Лицензія: Безкоштовно (персональне використання) та умовно-безплатне (пробний період 30 днів)

Система: Windows 7/8/10

Сайт: <https://www.avira.com/en/free-antivirus-windows>

Zillya! – антивірусний програмний продукт, розроблений українською компанією призначений для захисту комп'ютерів від різних видів шкідливих програм, таких як віруси, трояни, хробаки, шпигунські програми та інші загрози з Інтернету.

Програма має велику базу даних з підписами вірусів і евристичним аналізатором, що дозволяє виявляти нові шкідливі програми, які ще не внесені до бази даних. Крім того, Zillya! має функцію виявлення потенційно небезпечних додатків та блокування небажаних сайтів.

Програма працює на платформах Windows і Android і має різні версії для домашнього користування та бізнесу. Вона має простий і зручний інтерфейс, що дозволяє користувачам легко налаштувати рівень захисту свого комп'ютера або мобільного пристрою.

Розробник: Лабораторія Zillya! (Україна)

Лицензія: умовно-безплатне (пробний період 15 днів)

Система: Windows 7/8/10/XP, Android 2.0

Сайт: <https://zillya.ua/>

Norton 360 – комплексне програмне забезпечення для захисту комп'ютера. Воно включає в себе різні інструменти,

такі як антивірус, захист від шкідливих програм, перевірка безпеки Wi-Fi, захист від шахрайства та багато іншого.

Norton 360 також надає можливість зберігати та керувати паролями, захищати особисті дані та здійснювати резервне копіювання файлів. Він доступний як для персональних комп'ютерів, так і для мобільних пристроїв. Загалом, Norton 360 є потужним інструментом для захисту комп'ютера та особистої інформації від різних загроз в Інтернеті.

Розробник: Symantec Corporation (США)

Ліцензія: Безкоштовно немає, але є гарантія повернення коштів протягом 60 днів.

Система: Windows 7/8/10.

Сайт: <https://us.norton.com/products/norton-360>

F-Secure SAFE – програмне забезпечення для комп'ютерів, смартфонів та планшетів, яке забезпечує захист від різних видів онлайн загроз, таких як віруси, шпигунське програмне забезпечення та хакерські атаки.

F-Secure SAFE також має функції для безпеки приватності, такі як блокування небажаних дзвінків та повідомлень, а також фільтри для блокування шкідливого вмісту в Інтернеті; має функцію батьківського контролю.

F-Secure SAFE є продуктом фінської компанії F-Secure, яка спеціалізується на розробці програмного забезпечення для захисту від онлайн загроз.

Розробник: F-Secure (Фінляндія)

Ліцензія: Безкоштовно немає, але доступна пробна версія на 30 днів для трьох пристроїв.

Система: Windows 7/8/10/XP, macOS, Android, iOS.

Сайт: <https://www.f-secure.com>

Засоби VPN

Radmin VPN – безкоштовна програма для створення віртуальної приватної мережі (VPN), що дозволяє створювати безпечне з'єднання між комп'ютерами через Інтернет. За його допомогою можна легко підключати

комп'ютери до мережі і обмінюватися даними без ризику проникнення зовнішніх користувачів.

Radmin VPN має простий інтерфейс та засоби налаштування параметрів безпеки мережі, таких як шифрування даних та аутентифікація користувачів.

Завдяки Radmin VPN можна безпечно підключатися до віддалених комп'ютерів, наприклад, для доступу до робочого місця з дому або для гри в онлайн-ігри з друзями.

Розробник: Famatech (Британські Віргінські острови)

Ліцензія: Безкоштовно (персональне використання) та умовно-безплатне (пробний період 30 днів)

Система: Windows 7/8/10/11

Сайт: <https://www.radmin-vpn.com/ua/>

OpenVPN – віртуальна приватна мережа (VPN), яка забезпечує безпечне підключення до Інтернету шляхом шифрування даних та створення безпечного каналу зв'язку між комп'ютерами. OpenVPN є відкритим програмним забезпеченням, що може бути встановлено на різних операційних системах, включаючи Windows, Linux та MacOS.

Розробник: OpenVPN Inc. (США)

Ліцензія: умовно-безплатне (пробний період 30 днів)

Система: Windows 7/8/10

Сайт: <https://openvpn.net/>

ExpressVPN – послуга віртуальної приватної мережі, яка забезпечує високу конфіденційність та безпеку при користуванні Інтернетом. Це програмне забезпечення дозволяє користувачам змінювати свою мережеву адресу та шифрувати дані, які передаються через Інтернет, що робить їх недоступними для сторонніх осіб та підвищує рівень безпеки під час використання громадських мереж Wi-Fi. ExpressVPN також дозволяє користувачам отримувати доступ до заблокованих веб-сайтів та сервісів, які можуть бути недоступні у їхньому регіоні.

Розробник: Karpe Technologies (Британські Віргінські острови)

Лицензія: умовно-безплатне (пробний період 45 днів)

Система: Windows 7/8/10/11, macOS, Android, iOS, Linux.

Сайт: <https://www.expressvpn.com/>

Hotspot Shield – програмне забезпечення для створення віртуальної приватної мережі, яке шифрує дані користувача та забезпечує конфіденційність його онлайн-діяльностей шляхом забезпечення безпечного з'єднання з сервером Hotspot Shield, а також дозволяє отримувати доступ до заблокованого контенту.

Розробник: Karpe Technologies (США)

Лицензія: умовно-безплатне (пробний період 30 днів)

Система: Windows 7/8/10/11, macOS, Android, iOS, Linux.

Сайт: <https://www.expressvpn.com/>

Завдання до практичних занять

Тема 1: Основні поняття в області безпеки інформаційних технологій.

Завдання 1. Створіть текстовий документ, що містить відомості про історію розвитку інформаційної безпеки. Подайте знайдені відомості в текстовому документі у зручному вигляді (таблиці, схеми тощо).

Тема 2: Основні причини загострення проблеми забезпечення безпеки інформаційних технологій. Інформація та інформаційні відносини.

Завдання 1. Підготуйте міні-проекти про загрози безпеці: інтернет- шахрайство; спам-розсилки; бот-мережі (botnet), DDoS-атаки (Distributed Denial of Service), крадіжка коштів, «крадіжка особистості» (Identity Theft).

Завдання 2. Виконайте інтерактивну вправу «Інформаційна безпека в Інтернеті» <https://learningapps.org/4518841>

Завдання 3. Створіть ребус чи кросворд до термінів вивчених на уроці (скористатися генератором ребусів чи кросвордів).

Тема 3: Загрози безпеці інформації в автоматизованих системах.

Завдання 1. Складіть словник термінів (сленг програмістів, хакерів), що стосуються інформаційної безпеки.

Завдання 2. Виконайте інтерактивну вправу «Чинники загроз безпечіданих» <http://LearningApps.org/3944076>

Завдання 3. Виконайте інтерактивну вправу «Класифікація загроз безпеці» <https://learningapps.org/6278374>

Завдання 4. Виконайте інтерактивну вправу «Загрози безпеці та пошкодження даних у комп'ютерних системах» (розділ Захист інформації) <https://learningapps.org/5745146>

Тема 4: Технічні та програмні засоби добування інформації.

Завдання 1. Знайдіть відомості в Інтернеті та складіть список з 10 найбільш небезпечних комп'ютерних вірусів. Визначте, з якою метою вони були створені. Подайте знайдені відомості у вигляді презентації.

Завдання 2. Зробіть дослідження про типи шкідливих програм (virus Trojan, Backdoor, Dropper, Downloader, Tool, Adware, Dialer, Worm, Exploit, Rootkit), скористайтеся даними на сайті Zillya! (вірусна енциклопедія)

Тема 5: Практична робота «Використання програмних засобів для тестування та очищення ОС від вірусів та шкідливого ПЗ»

Мета: навчитися використовувати програмні засоби для тестування та очищення операційної системи від вірусів та шкідливого програмного забезпечення.

Хід роботи

Завдання 1. З'єднайте лініями назви різновидів вірусів із середовищами їхнього розповсюдження.

Завантажувальні віруси	У файлах документів
Файлові віруси	У виконуваних файлах
Мережні віруси	У мережі
Макровіруси	У тій самій області диска, що й файли ОС

Завдання 2. Позначте, які шкідливі дії можуть виконувати віруси.

- Сповільнювати роботу комп'ютера.
- Виводити з ладу процесор.
- Видаляти дані.
- Зменшувати обсяг вільного місця на диску.
- Призводити до помилок в роботі ОС.
- Блокувати роботу антивірусних програм.

Завдання 3. Вкажіть ознаки зараження комп'ютера вірусами чи троянами.

«Зависання» комп'ютера.

- Припинення роботи чи неправильна робота програм, що раніше функціонували нормально.
- Незвичний шум у системному блоці.
- Збільшення кількості файлів на диску, змінення їхніх обсягів, дати й часу створення.
- Періодична поява на екрані монітора недоречних повідомлень. Нестабільність з'єднання з Інтернетом.

Завдання 4. Запустіть на виконання антивірусну програму.

- Встановіть «Zillya!», Сайт: <https://zillya.ua/>
- Запустіть файл утиліти Zillya! на виконання (двічі клацніть по ньому лівою кнопкою мишки).

Завдання 5. Визначте за допомогою довідки, які операції виконує дана програма; до якого виду антивірусних програм її слід віднести.

- У вікні програми Zillya! (Рис. 1) вибрати кнопку «Налаштування».
- Зі списку вибрати «Сканування», «Інспектор» та «Вартовий».



Рис. 1

Завдання 6. Установіть такі значення параметрів сканування, інспектор та вартовий:

1. Дії над загрозами під час сканування – видаляти загрози.
2. Архівні файли – увімкнути.
3. Евристичний аналіз – увімкнути.
4. Завантажувальні сектори – увімкнути.
5. Вартовий – лікувати (в разі невдачі, помістити в карантин).

Хід виконання:

1. У вікні програми Zillya! (Рис. 1) вибрати кнопку «Налаштування».
2. У вкладці «Сканування», установити позначку

- прапорця (увімкнути) біля «Архівні файли», «Евристичний аналіз» та встановити «Дії над загрозами під час сканування» – видаляти загрози;
- У вкладці «Інспектор», встановити «Завантажувальні сектори» – увімкнути;
 - У вкладці «Вартовий», встановити «Вартовий» – лікувати (в разі невдачі, помістити в карантин).
 - Поверніться до головної сторінки.

Завдання 7. Виконайте антивірусну перевірку об’єктів папки Мої документи.

- У вікні програми Zillya! (Рис. 1), де кнопка «Швидке сканування» натисніть на стрілку яка знаходиться внизу.
- Виберіть «Вибіркове сканування»
- Установити позначку прапорця біля «Мої документи» (Рис. 2).
- Натиснути «Розпочати».



Рис. 2

Завдання 8. Дайте відповідь на запитання: Скільки об'єктів було перевірено? Чи були знайдені віруси?

Завдання 9. Зафіксувати звіт про роботу Zillya!

Завдання 10. Записати у зошит звіт.

Тема 6: Об'єкти захисту. Види заходів протидії загрозам безпеки.

Завдання 1. Створіть текстовий документ, що містить відомості про систему інформаційної безпеки. Подайте знайдені відомості в текстовому документі у зручному вигляді (таблиці, схеми тощо).

Тема 7: Правові основи забезпечення безпеки інформаційних технологій.

Завдання 1. Створіть інформаційний бюлетень із правилами комп'ютерного етикету.

Завдання 2. Виконайте інтерактивну вправу «Інформаційна етика» за посиланням <http://learningapps.org/watch?v=p502ii7et16>

Тема 8: Ідентифікація та аутентифікація користувачів. Розмежування доступу до ресурсів автоматизованих систем.

Завдання 1. Ознайомтеся з основними правилами створення надійного пароля на сайті [https://cyberukraine.in.ua/ \(#ПарольГраєРоль\)](https://cyberukraine.in.ua/#ПарольГраєРоль)

Завдання 2. Здійсніть перевірку пароля на сайті Zillya!: <https://zillya.ua/ru/check-password>

Завдання 3. За матеріалами Інтернет-ресурсу підготуйте бюлетень «Як придумати надійний пароль і де його зберігати».

Тема 9: Контроль цілісності програмних і інформаційних ресурсів. Виявлення атак.

Завдання 1. За матеріалами Інтернету підготуйте доповідь «Які є види мережевих атак?».

Тема 10: Практична робота «Використання засобів адміністрування ОС для налаштування прав користувачів щодо доступу до інформаційних ресурсів»

Мета: навчитися використовувати засоби адміністрування операційної системи для налаштування прав користувачів щодо доступу до інформаційних ресурсів

Хід роботи

Завдання 1. Виконати аналіз налаштувань захисту ПК.

Порядок виконання:

1. Перевірити, чи включений брандмауер.
2. Перевірити, чи налаштовано оновлення операційної системи.
3. Скільки облікових записів є на вашому робочому ПК? Які з них захищені паролем? Який тип вашого облікового запису?
4. Визначити, яке встановлене антивірусне ПЗ, як здійснюється оновлення антивірусної програми, які вона має можливості?
5. Які загрози втрати інформації на вашому комп'ютері можуть бути?
6. Отримані результати запишіть до звіту.

Завдання 2. Виконати аналіз роботи з паролями.

Порядок виконання:

1. Запустити браузер.
2. Відвідати ресурс <https://app.fortnotes.com>
3. Створіть власний акаунт. Проаналізуйте можливості даної служби. Запишіть основні можливості даної служби.
4. Створіть кілька паролів електронної пошти, соціальних мереж.
5. Відвідати ресурс <http://websecurity.com.ua/password>
6. Ознайомтесь з можливостями та призначенням ресурсу.
7. Введіть власні паролі для діагностики рівня її надійності.
8. Запишіть в звіт результати.

Завдання 3. Виконати синхронізацію даних.

Порядок виконання:

1. Запустити браузер
2. Ввійдіть в свій обліковий запис Google, відкрийте додаток Диск.
3. Виберіть вкладку Комп'ютери. На цій вкладці завантажте додаток Резервнекопіювання та синхронізація Windows.
4. Відкрийте додаток на ПК. Ввійдіть в свій обліковий запис.
5. Виберіть свою папку (ту, яку будете відновлювати) для синхронізації
6. Створіть текстовий документ та додайте кілька речень. Завантажте декілька зображень.
7. Перегляньте, чи синхронізуються дані.
Завдання 4. Напишіть звіт про виконану роботу.

Тема 11: Проблеми забезпечення безпеки в комп'ютерних системах і мережах. Засоби захисту мереж.

Завдання 1: Ознайомтеся з основними загрозами в мережі, скориставшись посиланнями:

1. посилання: Укртелеком (загрози в Інтернеті)
<https://ukrtelecom.ua/help/security/>
2. посилання: Zillya (Загрози в мережі Інтернет)
<https://zillya.ua/zagrozi-v-merezh%D1%96-%D1%96internet>

Завдання 2. Розробіть загальні рекомендації щодо захисту мереж.

Тема 12: Призначення, можливості, і основні захисні механізми міжмережевих екранів (брандмауерів).

Завдання 1. З'ясуйте, що таке файли cookies і навіщо їх видаляють. Як видалити куки? Створіть текстовий документ, уведіть відповідь.

Завдання 2. Практичне завдання: Навчитися працювати з поштою,брандмауером і на захищених сайтах.

1. Запустіть текстовий процесор і створіть новий документ.
2. Визначте, чи увімкнено брандмауер Windows на вашому комп'ютері. Зробіть скріншот вікна налаштування брандмауера та збережіть у створеному документі.
3. Запустіть браузер та відкрийте один із захищених сайтів. Зробіть скріншот вікна захищеного сайта та збережіть у документі.
4. Запишіть у документ пояснення, чому, на вашу думку, актуальним є використання захищеного каналу зв'язку.
5. Збережіть файл документа у відповідній папці.

Тема 13: Основні захисні механізми. Політика безпеки при доступі до мережі загального користування.

Завдання 1. Підготувати повідомлення на тему: «Віртуальні приватні мережі».

Тема 14: Системи аналізу вмісту поштового і веб-трафіку. Віртуальні приватні мережі (VPN).

Завдання 1. Створіть Google-презентацію на тему «ТОП 5 безкоштовних VPN в Україні. Переваги та недоліки».

Тема 15: Практична робота «Створення віртуальних приватних мереж».

Мета: навчитися використовувати віртуальні приватні мережі.

Хід роботи

Завдання. Створення вхідного підключення до VPN-сервера.

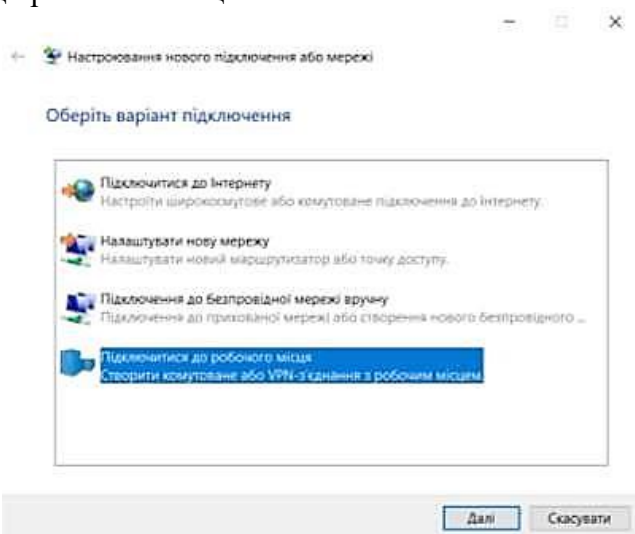
Для створення вхідного підключення до VPN-сервера необхідно зробити наступні кроки:

1) Виберіть «Панель керування -> Центр мережевих підключень і спільного доступу».



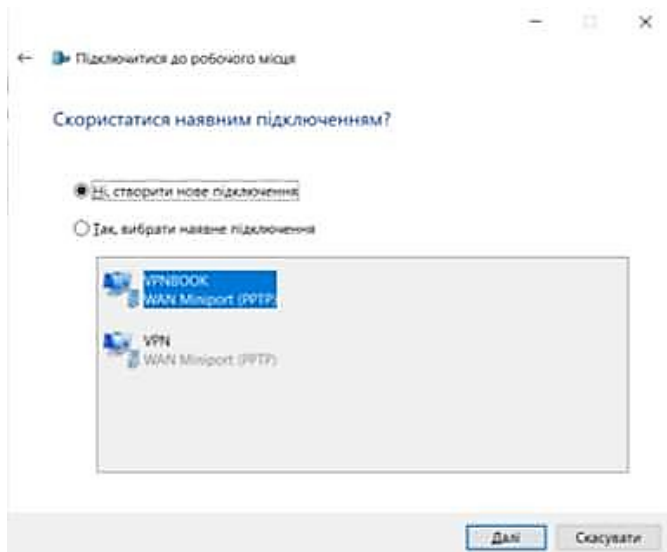
2) У меню «Центр мережевих підключень і спільного доступу» виберіть «Настроїти нове підключення або мережу» для запуску майстра нових підключень.

3) В меню «Настроювання нового підключення або мережі» оберіть варіант підключення «Підключення до робочого місця».

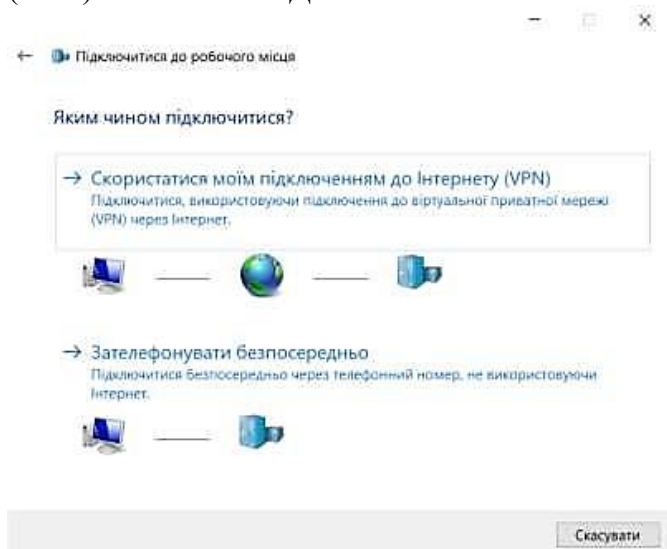


4) Натисніть «Далі».

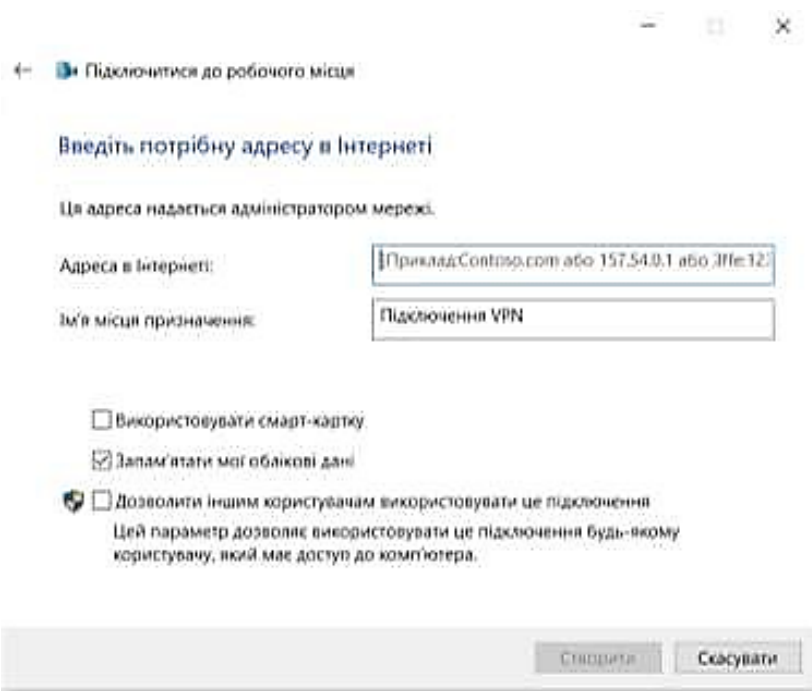
5) У вікні «Скористатися наявним підключенням?» виберіть «Ні, створити нове підключення» і натисніть «Далі».



- 6) У вікні «Яким чином підключитися?» виберіть «Скористатися моїм підключенням до Інтернету (VPN)» і натисніть «Далі».



- 7) У вікні «Підключитися до робочого місця» введіть IP-адресу: ca222.vpnbook.com і ім'я підключення «VPN».

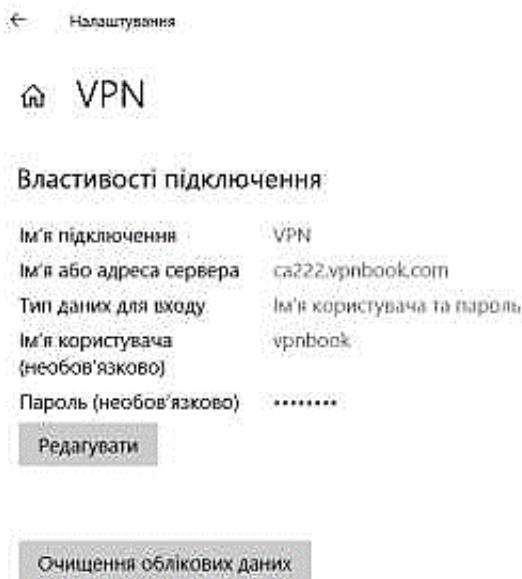


8) Зайдіть в «Налаштування» -> «Мережа й Інтернет» -> «Мережа VPN».

9) Виберіть мережу VPN «VPN».



10) Натисніть на «Додаткові параметри», потім виберіть «Редагувати».



- 11) У вікні «Редагування підключення до мережі VPN» заповніть ім'я користувача «vpnbook» та пароль «dd4e58m», і натисніть «Зберегти».

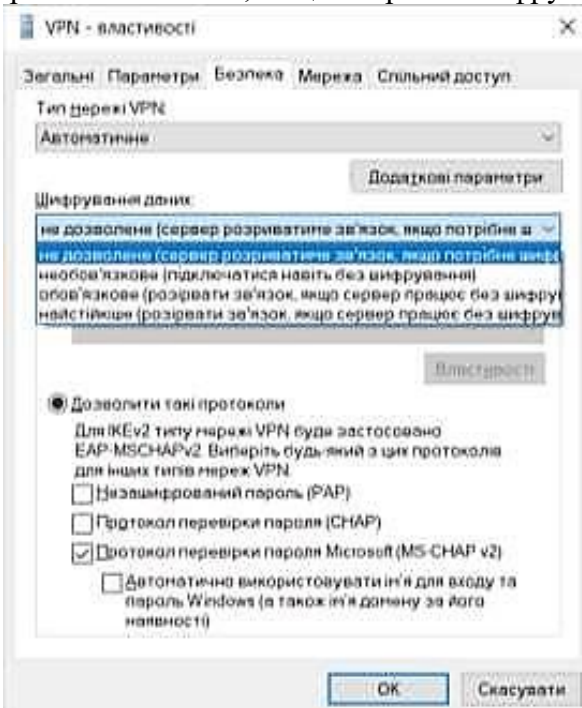


- 12) Поверніться до вікна «Центр управління мережами і загальним доступом» зліва виберіть посилання «Змінити налаштування адаптера»:

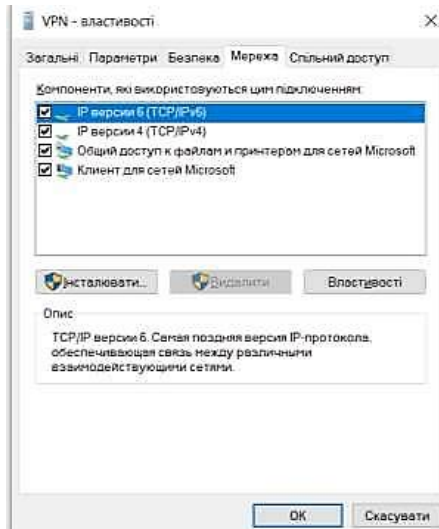


13) У вікні «Мережеві підключення» виберіть створене підключення, натисніть правою кнопкою миші і у вкладці натисніть «Властивості».

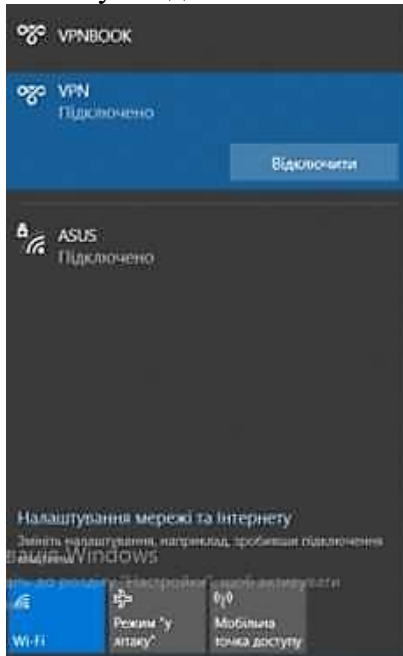
14) Виберіть вкладку «Безпека». В меню шифрування даних встановіть параметр «Не дозволене (сервер розриватиме зв'язок, якщо потрібно шифрування)».



15) Виберіть вкладку «Мережа». В полі «Компоненти, що використовуються цим підключенням» зніміть прапорець «Протокол Інтернет версії 6 (TCP IP 6)», виберіть «Протокол Інтернет версії 4 (TCP IP 4)» і натисніть «Властивості».



- 16) Запустіть створене підключення у вікні мережевих підключень або за допомогою значка «Мережа» на панелі завдань (де годинник !!!). У додатковому вікні натисніть кнопку «Підключення».



Якщо всі налаштування зроблені правильно, то Ви побачите повідомлення про вдале підключення.

Тема 16: Практична робота «Налаштування засобів системного міжмережевого екрану (брандмауера)».

Мета: формувати вміння практичного застосування знань, умінь і навичок; формувати й удосконалювати вміння й навички у стандартах прямого й успішного руху до накресленої мети.

Хід роботи

Завдання 1. Налаштування брандмауера.

1. Відкрити «Налаштування -> Система й безпека».
2. Перевірити стан брандмауера. (Брандмауер повинен бути увімкнений для приватної мережі і мережі спільного використання).
3. Переглянути програми, яким дозволено обмінюватися даними через брандмауер.

Завдання 2. Облікові записи.

1. Відкрити «Налаштування -> Облікові записи», оберіть родина та інші користувачі.
2. Додати користувача на цей ПК/в діалоговому вікні обираємо «У мене немає облікових даних цієї особи».
3. Обираємо «Додати користувача без облікового запису Microsoft».
4. Вводимо облікові дані для користувача.
5. В переліку облікових записів обираємо необхідний та надаємо тип «Стандартний користувач».
6. Перевіряємо перелік облікових записів.

Завдання 3. Резервне копіювання.

1. Відкрити «Налаштування -> Оновлення та захист».
2. Обираємо пункт «Резервне копіювання».
3. Вставляємо носій інформації (може бути флеш-накопичувач).
4. Обираємо його з переліку дисків.

5. Обрати, які папки потрібно для резервного копіювання та вказати інтервал копіювання (для цього обираємо інші параметри).
6. Змінити параметр інтервалу резервного копіювання на кожні 3 години.
7. Додати свою папку для резервного копіювання, для цього натискаємо «Додати папку», в діалоговому вікні обираємо свою папку.
8. Всі інші папки видаляємо, для резервного копіювання потрібна тільки робоча папка.

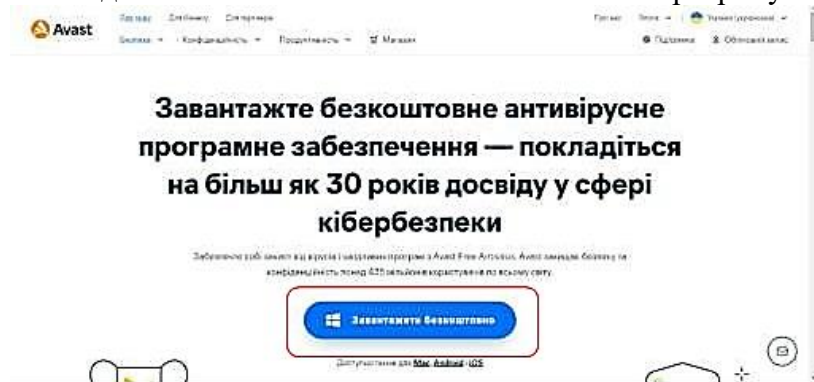
Тема 17: Антивірусні засоби захисту.

Мета: навчитися працювати з антивірусною програмою «Avast Free Antivirus».

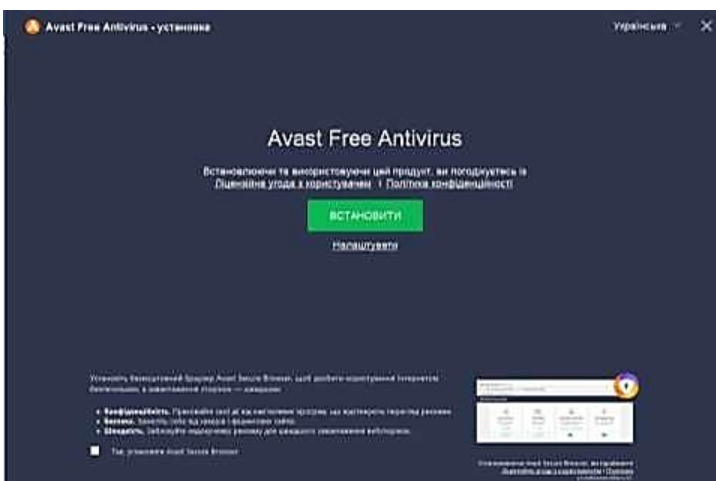
Хід роботи

Завдання 1. Установіть антивірусну програму «Avast Free Antivirus» із Інтернету і установіть цю програму.
Порядок виконання:

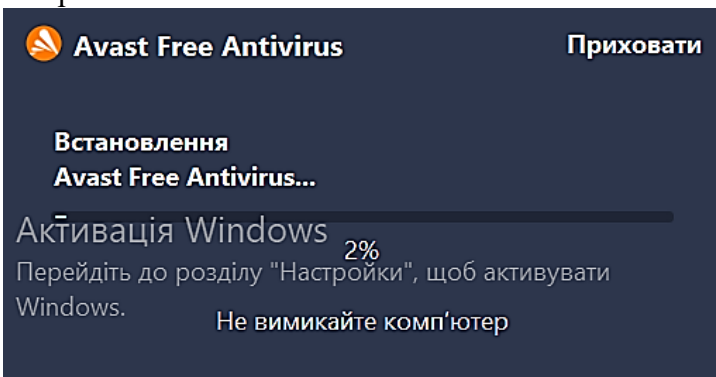
1. Зайдіть на сайт www.avast.com і завантажте програму.



2. Установіть файл `avast_free_antivirus_setup_online`.
3. Подвійним клацанням запустіть встановлену програму даного файлу. Відкрийте вказаному діалоговому вікні.
4. Після натиснення кнопки «Так» відкриється вікно:



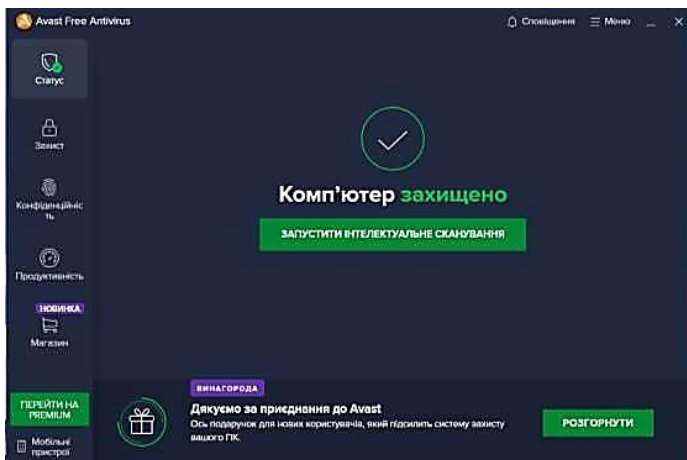
5. Натисніть на кнопку «Встановити».
6. Почнеться процес встановлення і після цього певного кроку буде виведено повідомлення про успішне завершення.



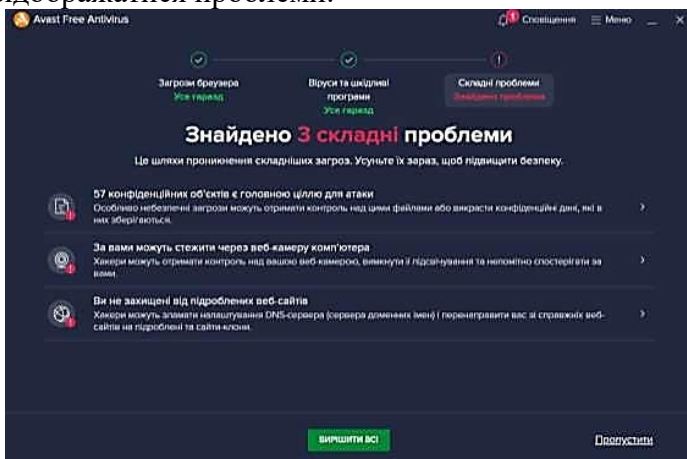
7. Перезавантажте комп'ютер або ноутбук. Після перезавантаження в правому кутку екрана (на панелі задач) з'явиться ярлик програми «Avast Free Antivirus».

Завдання 2. Робота з антивірусною програмою «Avast Free Antivirus».

1. Запустіть програму, натисніть по відповідному значку на робочому столі або на панелі задач. Відкрийте основне вікно програми.



2. Натисніть на кнопку «Запустити інтелектуальне сканування». Почнеться процес сканування.
3. По завершенні сканування в програмі буде відображатися проблеми.



4. Для дозволу виявлених проблем натисніть на кнопку «Вирішити всі».
5. Вам буде запропоновано варіанти як вирішити цю проблему.
6. По закінченню перевірки закрийте вікно програми.

Завдання 3. Створіть пам'ятку «Алгоритм дій, якщо Ваш ПК інфіковановірусом».

Підсумковий тест з вибіркового модуля «Інформаційна безпека»²

Одна правильна відповідь

1. Як називається навмисне спотворена інформація?
а) Дезінформація
б) Інформаційний потік
в) Достовірна інформація
г) Перестає бути інформацією
2. Захист інформації забезпечується застосуванням антивірусних засобів
а) Не завжди
б) Ні
в) Так
3. При реєстрації нового облікового запису, який пароль Ви будете використовувати?
а) Пароль, який легко запам'ятати (дата народження, прізвище тощо)
б) Пароль, який використовую у всіх інших облікових записах
в) Пароль, який за умовчанням запропонувала система або був висланий під час реєстрації
г) Унікальний складний пароль (з цифрами, символами різного регістру, спецсимволами)
4. Переважна більшість загроз інформаційної безпеки припадає на:
а) Троянські програми
б) Хробаки
в) Шпигунські програми
г) всі три варіанти правильні
5. Які дані про користувачів відстежуються соціальними мережами (Instagram, Telegram, Viber та ін.)?
а) Місце користувача

² Правильні відповіді відмічено жирним шрифтом.

- б) Час користування сервісами
- в) Дані про пристрій користувача
- г) Інформація про те, що подобається користувачу**

Декілька правильних відповідей

6. Основні загрози доступності інформації:
- а) хакерська атака
 - б) руйнування чи пошкодження приміщень**
 - в) відмова програмного та апаратного забезпечення**
 - г) перехоплення даних
 - д) ненавмисні помилки користувачів**
 - е) зловмисне зміна даних
7. Які дані можна надсилати своїм друзям та знайомим у соціальних мережах та месенджерах?
- а) Фотографії сторінок паспорта**
 - б) PIN-коди або CVV2/CVC2 коди банківських карток**
 - в) Секретні слова та/або відповіді на спеціальні секретні питання, які використовувалися для ідентифікації в обліковому записі
 - г) Свої ім'я та прізвище, життєві історії
8. Вам надійшов лист від адміністрації соціальної мережі про те, що Ваш обліковий запис зламаний. Щоб його відновити, автори листа просять надіслати пароль. Як слід вчинити у цій ситуації?
- а) Відповісти на лист і надіслати пароль
 - б) Перезавантажити комп'ютер
 - в) Проігнорувати лист, не відповідати на нього**
 - г) Видалити лист, зайти в соціальну мережу та поміняти пароль**

Відповідність.

9. Встановіть відповідність рівня безпеки та його описом:

1. Особистий	А) Оскільки все більше цифрової інформації збирається та розповсюджується, її захист стає ще важливішим на рівні уряду, де на
--------------	---

	кону стоїть національна безпека, економічна стабільність, безпека та добробут громадян.
2. Корпоративний	Б) На особистому рівні вам необхідно захищати свій обліковий запис, дані та електронні пристрої.
3. Урядовий	В) На корпоративному рівні відповідальність кожного співробітника полягає в захисті репутації організації, її даних та клієнтів.
	Г) Будь-яка інформація, яка може бути використана для вашої ідентифікації, і вона може існувати як офлайн, так і онлайн.

Відповідь: 1 – Б, 2 – В, 3 – А

10. Встановіть відповідність між термінами та їхніми означеннями:

1. Шифрування	А) Зберігання даних локально означає повний контроль над ними.
2. Домашня мережа	Б) В основному використовуються для з'єднання різних сегментів мережі разом, вони зазвичай також забезпечують основні можливості фільтрації трафіку.
3. Маршрутизатори	В) Набір сигнатур трафіку, які відповідають шкідливому трафіку і атакам та блокують зв'язок при співпаданні.
	Г) Перетворення інформації у форму, непридатну для сприйняття неавторизованою стороною.

Відповідь: 1 – Г, 2 – А, 3 – В

Дати відповідь на запитання.

11. Що таке кібербезпека?

Кібербезпека – це постійні зусилля щодо захисту окремих осіб, організацій та урядів від цифрових атак шляхом захисту мережевих систем і даних від несанкціонованого використання або шкоди.

12. Яка основна мета кібервійни?

Основна мета кібервійни – отримати перевагу над супротивниками, незалежно від того, чи є вони державами або конкурентами.

13. Чому внутрішня загроза безпеці може призвести до більшої шкоди для організації, ніж зовнішня загроза безпеці?

Внутрішні користувачі мають прямий доступ до пристроїв інфраструктури.

14. В чому полягає мотивація хакера «у білому капелюсі»?

Точне налаштування мережних пристроїв для покращення їх ефективності та продуктивності.

15. Як розшифровується DoS?

Атаки відмови в обслуговуванні

16. Що таке ботнет?

Ботнет – це група ботів, з'єднаних через Інтернет, які можуть контролюватися одним зловмисником або групою.

Заповнити пропуск.

17. Індивідуальний профіль користувача на сайті соціальної мережі є прикладом ... особистості.

онлайн

18. Порухення безпеки також може мати руйнівний вплив на конкурентоспроможність організації, особливо якщо ... можуть отримати у свої руки конфіденційні документи, комерційну таємницю та інтелектуальну власність.

хакери

19. Зловмисник будує мережу (ботнет) із заражених вузлів, які називаються ..., які контролюються системами керування.

зомбі

20. Системи віртуальних приватних мереж VPN дозволяють віддаленим співробітникам використовувати безпечний ... тунель зі свого мобільного комп'ютера та безпечно підключатися до мережі організації.

зашифрований

21. «Відкритий» стан порту означає, що до служби, яка працює в мережі, можна отримати ... з інших мереж, і якщо служба дійсно містить вразливість, вона може бути ... зловмисником, який потенційно може отримати доступ до комп'ютерів у мережі.

доступ, експлуатована

Критерії оцінювання тестування.

Оцінка	Неправильно	Правильно
1	21	0
2	17-20	1-4
3	13-16	5-8
4	12	9
5	11	10
6	9-10	11-12
7	8	13
8	7	14
9	5-6	15-16
10	4	17
11	2-3	18-19
12	0-1	20-21

Вибірковий модуль «Веб-технології» (35 год.)

Укладач Б.М.Теренчин

Календарно-тематичний план

складений відповідно до чинної програми: «Інформатика 10 – 11 класи (рівень стандарту). Програма для загальноосвітніх навчальних закладів». Затверджено Наказом Міністерства освіти і Науки України від 23.10.2017 № 1407

№ уроку	Тема уроку	Примітки
Тема 1. Напрямки та інструменти веб-дизайну		
1	Правила поведінки і безпеки життєдіяльності (БЖ) в комп'ютерному класі. Поняття веб-дизайну. Основні тренди веб-дизайну.	
2	Інструктаж з БЖД. Типи сайтів та їх особливості. Цільова аудиторія.	
3	Інструктаж з БЖД. Інформаційна структура сайту.	
4	Інструктаж з БЖД. Інструменти веб-розробника.	
5	Інструктаж з БЖД. План розробки сайту. Макет інформаційної структури сайту.	
6	Інструктаж з БЖД. <i>Практична робота №1: «Проектування структури сайту».</i>	
Тема 2. Проектування та верстка веб-сторінок		
7	Інструктаж з БЖД. Мова гіпертекстової розмітки. Гіпертекстовий документ та його елементи.	
8	Інструктаж з БЖД. Текстові елементи веб-сторінки, теги та їх атрибути. Гіперпосилання.	
9	Інструктаж з БЖД. Таблиці та списки на веб-сторінках.	
10	Інструктаж з БЖД. <i>Практична робота №2: «Створення веб-сторінки за допомогою «Мови гіпертекстової розмітки».</i>	

№ уроку	Тема уроку	Примітки
11	Інструктаж з БЖД. Каскадні таблиці стилів. Стильове оформлення сторінок.	
12	Інструктаж з БЖД. Позичіювання елементів веб-сторінки за допомогою стилів.	
13	Інструктаж з БЖД. <i>Практична робота №3</i> : «Створення веб-сторінки з використанням каскадних аркушів стилю»	
14	Інструктаж з БЖД. Проектування та верстка веб-сторінок.	
15	Інструктаж з БЖД. Адаптивна верстка.	
16	Інструктаж з БЖД. Кросбраузерна оптимізація сторінок сайту.	
Тема 3. Графіка та мультимедіа для веб-середовища		
17	Інструктаж з БЖД. Використання зображень у веб-документах.	
18	Інструктаж з БЖД. Створення GIF-анімації.	
19	Інструктаж з БЖД. Карти посилань.	
20	Інструктаж з БЖД. Розміщення та відтворення на веб-сторінках мультимедійних даних. Формати аудіо- та відеофайлів.	
21	Інструктаж з БЖД. Дотримання авторських прав та ліцензій на використання графічних зображень та засобів мультимедіа на веб-сторінках.	
22	Інструктаж з БЖД. <i>Практична робота №4</i> : «Використання графічних, анімаційних та мультимедійних елементів на веб-сторінках».	
Тема 4. Веб-програмування		
23	Інструктаж з БЖД. Поняття об'єктної моделі документа. Веб-програмування та інтерактивні сторінки.	
24	Інструктаж з БЖД. Використання форм. Основні елементи форми.	
25	Інструктаж з БЖД. Валідація та збереження даних форм.	

№ уроку	Тема уроку	Примітки
26	Інструктаж з БЖД. <i>Практична робота №5: «Створення та налагодження інтерактивної веб-сторінки з використанням форм та веб-програмування»</i>	
27	Інструктаж з БЖД. Хостинг сайту. Взаємодія клієнт-сервер.	
28	Інструктаж з БЖД. <i>Практична робота №6: «Розміщення сайту на сервері»</i>	
Тема 5. Основи дизайну та просування веб-сайту		
29	Інструктаж з БЖД. Правила ергономічного розміщення відомостей на веб-сторінці.	
30	Інструктаж з БЖД. Пошукова оптимізація та стратегії просування веб-сайтів.	
31	Інструктаж з БЖД. <i>Практична робота №7: «Оцінка сайту. Просування сайту»</i>	
32	Інструктаж з БЖД. Проектна діяльність. Розбиття на групи. Вибір теми проекту. Розробка плану проекту.	
33	Інструктаж з БЖД. Робота над проектом.	
34	Інструктаж з БЖД. Захист проекту.	
35	Інструктаж з БЖД. Повторення і систематизація навчального матеріалу за рік.	

Список рекомендованої літератури (для вчителя):

1. Веб-технології та веб-дизайн : навч. посібник / О. Г. Трофименко, О. Б. Козін, О. В. Задерейко, О. Є. Плачінда. Одеса : Фенікс, 2019. 284 с.

Книга містить рекомендації щодо використання графічних редакторів, кольорової гами, типографіки та інших аспектів дизайну.

2. Мосіюк О. О. WEB-технології. Частина 1. Верстка. Житомир: Вид-во ЖДУ ім. Івана Франка, 2020. 56 с.

Ця книга, що охоплює основи HTML, CSS, допоможе вчителям ознайомитись з основами веб-дизайну та впровадити їх у навчальний процес.

3. Web-технології та Web-дизайн: застосування мови HTML для створення електронних ресурсів : навч. посіб. / І. Л. Бородкіна, Г. О. Бородкін. Київ: Видавництво Ліра-К, 2020. 212 с.

Книга спрямована на вчителів, які бажують ознайомити своїх учнів з основами веб-розробки. Вона містить пояснення ключових концепцій HTML, CSS та JavaScript, а також навчальні матеріали та практичні завдання.

4. Лясин, Д.Н., Абрамова О.Ф. Динамическое формирование HTML– документов на стороне сервера: методические указания / Д.Н. Лясин; ВПИ (филиал) ВолгГТУ. Волгоград, 2018. 29 с.
5. Вайнман Л., Вайнман В. Динамический HTML. Руководство разработчика Web-сайтов. К.: ООО «ТИД ДС», 2001. 464 с.
6. Ніколаєнко О.Ю., Кравченко С.М., Вернигоренко С.А. Використання HTML та JavaScript для створення WEB-документів: Посібник для студ.. К. : НПУ ім. М. П. Драгоманова, 2003. 112 с.

Список рекомендованої літератури для учнів

1. Інформатика (профільний рівень) : підруч. для 11 кл. закл. загал. серед. освіти / В. Д. Руденко, Н. В. Речич, В. О. Потієнко. Харків : Вид-во «Ранок», 2019. 256 с.
2. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закл. / [О.О. Бондаренко, В.В. Ластовецький, О.П. Пилипчук, Є.А. Шестопалов]. Харків : Вид-во «Ранок», 2022. 237 с.
3. Інформатика (профільний рівень) : підруч. для 11 кл. закл. загал. серед. освіти / В. Д. Руденко, Н. В. Речич, В. О. Потієнко. Харків : Вид-во «Ранок», 2019. 256 с.
4. Морзе Н. В. Інформатика (рівень стандарту): підруч. для 10 (11) кл. закладів загальної середньої освіти / Н. В. Морзе, О. В. Барна. К.: УОВЦ «Оріон», 2018. 240 с.

Програмне забезпечення для створення комп'ютерних анімацій

- ✓ *Visual Studio Code* – безкоштовне розширюване середовище розробки (IDE), яке надає потужні інструменти для роботи з веб-технологіями, підтримує багато мов програмування, має вбудовану консоль, підсвічування синтаксису, автодоповнення та багато інших корисних функцій.
- ✓ *Atom* – популярний безкоштовний текстовий редактор для веб-розробників. Він має багато розширень і тем оформлення, засоби налаштування та зручний інтерфейс.
- ✓ *Brackets* – безкоштовний текстовий редактор для веб-розробки. Він має вбудоване підсвічування синтаксису, засоби прямого перегляду змін в реальному часі та зручні інструменти для роботи з HTML, CSS та JavaScript.
- ✓ *Sublime Text* – популярний серед веб-розробників текстовий редактор, який має інтуїтивний і легкий у використанні інтерфейс, підсвічування синтаксису, автодоповнення та широкі можливості налаштування.

Практичні завдання до уроків

Практична робота №2 “Створення веб-сторінки за допомогою «Мови гіпертекстової розмітки».

Завдання

1. Створити на робочому столі папку та назвати її власним прізвищем та ім'ям.
2. У власні папці створити два html документа та назвати «Список», «Таблиця».
3. У документі «Список.html» реалізувати наступний список:

Розклад уроків

Понеділок

- Математика

- Фізкультура
- Вівторок
- Українська мова
 - Математика
- Середа
- Фізика
 - Хімія
- Четвер
- Історія
 - Англійська мова
- П'ятниця
- Англійська мова
 - Українська мова

4. У документі «Таблиця.html» реалізуйте наступне завдання:

<i>Понеділок</i>	<i>Вівторок</i>	<i>Середа</i>	<i>Четвер</i>	<i>П'ятниця</i>
Математика	Українська мова	Фізика	Історія	Англійська мова
Фізкультура	Математика	Хімія	Англійська мова	Українська мова

5. За допомогою головної сторінки, яка буде створена у вигляді головного меню, зв'яжіть сторінки «Список» та «Таблиця», надавши користувачу змогу обрати тип відображення розкладу. Головну сторінку оформіть на свій смак, встановивши фон сторінки і налаштувавши колірну систему тексту та гіперпосилань.
6. Результат роботи відправити вчителю на ел.адресу *електрона.пошта@домен*

Підказка до завдання 4 для учнів, які мають слабкі знання:

```

<!DOCTYPE html>
<html>
<head>
  <title>Розклад уроків</title>
  <style>
    table {

```

```

border-collapse: collapse;
width: 100%;
}

th, td {
border: 1px solid black;
padding: 8px;
text-align: center;
}

th {
background-color: #f2f2f2;
}
</style>
</head>
<body>
<table>
<tr>
<th>Понеділок</th>
<th>Вівторок</th>
<th>Середа</th>
<th>Четвер</th>
<th>П'ятниця</th>
</tr>
<tr>
<td>Математика</td>
</tr>
<tr>
<td>Фізкультура</td>
</tr>
<!-- Додайте решту рядків з інформацією про розклад -->
</table>
</body>
</html>

```

Практична робота №3 «Створення веб-сторінки з використанням каскадних аркушів стилю»

Завдання

1. На робочому столі створіть папку на Ваше прізвище.
2. За допомогою шаблону реферату, створіть веб-сторінку, використовуючи наступні зразки. Для задання стилів можна використовувати лише параметри додавання фону сторінки та колірні схеми для активних, пасивних та використаних посилань, а також тексту загалом. Усі інші форматування, такі як зміна розмірів, підкреслення посилань та позиціонування, будуть описані в файлі style.css, який буде підключений до кожного документу. Щоб абсолютно вирівняти текст по центру на сторінках 1, 2, 3, 4, буде використана таблиця з адаптивною шириною, яка займає 60% від ширини сторінки. Також буде використаний тег <center>

Шаблон реферату->

html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Реферат про якість інформації</title>
```

```
<link rel="stylesheet" type="text/css" href="style.css">
```

```
</head>
```

```
<body>
```

```
<div class="container">
```

```
<h1>Реферат про якість інформації</h1>
```

```
<table>
```

```
<tr>
```

```
<td>
```

```
<h2>Розділ 1</h2>
```

```
<p>Текст розділу 1...</p>
```

```
</td>
```

```
</tr>
```

```
</div>
```

```

        <td>
            <h2>Розділ 2</h2>
            <p>Текст розділу 2...</p>
        </td>
    </tr>
</tr>
<tr>
    <td>
        <h2>Розділ 3</h2>
        <p>Текст розділу 3...</p>
    </td>
</tr>
<tr>
    <td>
        <h2>Розділ 4</h2>
        <p>Текст розділу 4...</p>
    </td>
</tr>
</table>
</div>
</body>
</html>

```

CSS(style.css)

```

body {
    background-color: #f0f0f0;
    color: #333;
}

```

```

.container {
    margin: 0 auto;
    width: 60%;
    padding: 20px;
    background-color: #fff;
}

```

```

h1 {
    text-align: center;
}

```

```

}

h2 {
  color: #555;
}

table {
  width: 100%;
}

td {
  padding: 10px;
}

a:link, a:visited {
  color: #007bff;
  transition: color 0.3s;
}

a:hover, a:active {
  color: #0056b3;
}

p {
  line-height: 1.5;
  text-align: justify;
}

```

3. Збережіть результат виконання у вашу власну папку. Повідомте вчителю про виконання завдання.

Практична робота №4 «Використання графічних, анімаційних та мультимедійних елементів на веб-сторінках».

Завдання

1. Створити веб-сторінку за зразком. Використати таблицю 4x4, в комірках якої розмістити фото, а під ними відповідні підписи:

Курорти Карпат



[Драгобрат](#)

[Славське](#)

[Замки Львівщини](#)



[Коломия](#)



[Мукачево](#)



[Буковель](#)

2. Створити гіперпосилання на деяких підписах і на деяких фото (гіперпосилання на веб-сторінки відповідних курортів Карпат).

Курорти Карпат



[Драгобрат](#)

[Славське](#)

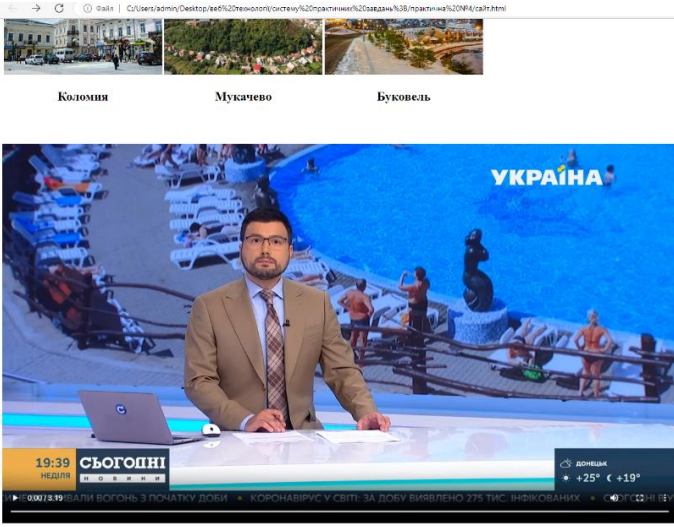
[Замки Львівщини](#)

Драгобрат — назва міста в області Львівщини, розташоване в долині річки Дрогобич. Це місто є одним з найбільш розвинених туристичних центрів у регіоні. Воно має багату історію та культурну спадщину. У місті знаходяться численні пам'ятники архітектури, музеї та парки. Драгобрат є популярним курортом завдяки своїй природній красі та історичним цінностям.

Славське — це курортне місто в області Львівщини, розташоване в долині річки Славське. Це місто є одним з найбільш розвинених туристичних центрів у регіоні. Воно має багату історію та культурну спадщину. У місті знаходяться численні пам'ятники архітектури, музеї та парки. Славське є популярним курортом завдяки своїй природній красі та історичним цінностям.

Замки Львівщини — це комплекс пам'яток архітектури в області Львівщини, розташований в долині річки Львів. Це комплекс пам'яток архітектури, який складається з кількох замків, побудованих в різні часи. Замки Львівщини є одним з найбільш розвинених туристичних центрів у регіоні. Вони мають багату історію та культурну спадщину. У замках знаходяться численні пам'ятники архітектури, музеї та парки. Замки Львівщини є популярним курортом завдяки своїй природній красі та історичним цінностям.

3. *Додаткове завдання:* розмістити відео на цій же веб-сторінці.



Примітка. Якщо учням ця тема складна, то варто запропонувати шаблон для заповнення:

```

<!doctype html>
<html lang="uk" >
<head>
<meta charset="UTF-8">
<title>Графіка</title>
</head>
<body>
<H2>Курорти Карпат </H2>
<table>
<tr>
<td><a
href="https://uk.wikipedia.org/wiki/%D0%94%D1%80%
D0%B0%D0%B3%D0%BE%D0%B1%D1%80%D0%B0
%D1%82"></a> </td>
<td><a href="https:// " ></a></td>

```

```

<td><a href="https://      "></a></td>
</tr>
<tr>
<td><center><H2><a
href="https://uk.wikipedia.org/wiki/%D0%94%D1%80%
D0%B0%D0%B3%D0%BE%D0%B1%D1%80%D0%B0
%D1%82">Драгобрам</a> </H2></center></td>
<td><center><H2><a href="https://      ">Славське</a>
</H2></center></td>
<td><center><H2><a href="https://      ">Замки
Львівщини</a> </H2></center></td>

</tr>
<tr>
<td><a href="https://      "></a></td>
<td><a href="https://      "></a></td>
<td><a href="https://      "></a></td>
</tr>
<tr>
<td><center><H2><a href="https://      " >Коломия</a>
</H2></center></td>
<td><center><H2><a href="https://      ">Мукачєво</a>
</H2></center></td>
<td><center><H2><a href="https://      ">Буковель</a>
</H2></center></td>
</tr>
</table>

<br><br><br>
<video src="video.mp4" width="350" controls</video>

</body></html>

```

Практична робота №5 «Створення та налагодження інтерактивної веб-сторінки з використанням форм та веб-програмування»

Завдання:

За поданим зразком створити тест на 12 питань з Інформатики чи Математики.

```
<HTML>
<HEAD>
<TITLE>Тест</TITLE>
<SCRIPT language=JavaScript>
    function check_me()
    {
        var count=0
        with(document.test)
        {
            if (!Q1[0].checked&&!Q1[1].checked&&!Q1[2].checked)
                {count+=1};
            if (!Q2[0].checked&&!Q2[1].checked&&!Q2[2].checked)
                {count+=1};
            if (!Q3[0].checked&&!Q3[1].checked&&!Q3[2].checked)
                {count+=1};

            if (count>0) {alert("Ви виконали не всі завдання.
                Перевірте себе.")}
            else
                {count=0;
                if (Q1[2].checked) {count+=1};
                if (Q2[1].checked) {count+=1};
                if (Q3[0].checked) {count+=1};
                if (count==0) {count=1};
                test.s1.value="Ваша оцінка "+count+"."; }
        }
    }
</SCRIPT>

</HEAD>
<BODY>
```

```

<FORM name=test>
Коли був випущений перший iPhone ? <BR>
<INPUT type=radio name=Q1> 2009 <BR>
<INPUT type=radio name=Q1>2007 <BR>
<INPUT type=radio name=Q1> 2011 <BR>

коли була створена вікіпедія? <BR>
<INPUT type=radio name=Q2> 2001; <BR>
<INPUT type=radio name=Q2> 2000;<BR>
<INPUT type=radio name=Q2> 1999. <BR>

скільки байтів в одному кілобайті?<br>
<INPUT type=radio name=Q3> 1024 <BR>
<INPUT type=radio name=Q3>256 <BR>
<INPUT type=radio name=Q3> 8 <BR>

<TEXTAREA name=s1 rows=1 cols=20> </TEXTAREA> <BR>
<INPUT onclick=check_me() type=button value="Показати
результат">
<INPUT type=reset value="Розпочати спочатку">
</FORM>

</BODY>
</HTML>

```

Практична робота №6 «Розміщення сайту на сервері»

Завдання

1. Зареєструватися на безкоштовному конструкторі сайтів (наприклад, Wix.com тощо).
2. Створити веб-сторінки з теми :
 - “Сім чудес України”;
 - “Моя Україна”;
 - “Сім чудес світу”;
 - “Видатні українці”;
 - “Джерела інформації”;
 - “Перші комп’ютери”;
 - “Здобутки українців у сфері інформатики”.

3. Зв'язати гіперпосиланнями кілька сторінок, а саме, обов'язково має бути сторінка про автора сайту, тобто його дані (ім'я, прізвище тощо) та посилання на джерела інформації.
4. Перевірте роботу сайту через браузер, використовуючи ім'я власного сайту.
5. Створити електронний лист на тему «Мій сайт» з текстом, що містить ПІБ виконавця, номер групи, URL власного сайту. Надіслати лист на адресу викладача *електрона.пошта@домен*.
6. Закрийте всі відкриті вікна.
7. Повідомте вчителю про завершення роботи.

Приклад сайту: <https://bogdanterenchin.wixsite.com/ldas>

Тести до практичних робіт

Тест до практичної роботи № 3

1. Що означає абревіатура HTML?
 - а) Hyper Text Markup Language
 - б) Hyperlinks and Text Markup Language
 - в) Home Text Markup Language
 - г) Hyperlinks and Technical Markup Language
2. Який елемент використовується для створення заголовку сторінки в HTML?
 - а) <header>
 - б) <h1>
 - в) <title>
 - г) <head>
3. Який тег використовується для створення посилання в HTML?
 - а) <a>
 - б) <link>
 - в) <p>
 - г) <div>
4. Як правильно вставити зображення на веб-сторінку в HTML?
 - а)

- б) `<image src="image.jpg" alt="Image">`
 - в) ``
 - г) ``
5. Яким елементом в HTML визначається нумерований список?
- а) ``
 - б) ``
 - в) ``
 - г) `<dl>`
6. Який тег використовується для встановлення основного заголовку сторінки в HTML?
- а) `<title>`
 - б) `<head>`
 - в) `<h1>`
 - г) `<header>`
7. Як встановити жирний шрифт для тексту в HTML?
- а) ``
 - б) `<bold>`
 - в) ``
 - г) ``
8. Як встановити курсивний шрифт для тексту в HTML?
- а) `<i>`
 - б) `<italic>`
 - в) ``
 - г) `<c>`
9. Як вставити горизонтальну лінію в HTML?
- а) `<line>`
 - б) `<hr>`
 - в) `<hrline>`
 - г) `<horizontal>`
10. Який тег використовується для створення таблиці в HTML?
- а) `<table>`
 - б) `<tr>`
 - в) `<td>`
 - г) `<th>`
11. Як правильно встановити фоновий колір для елемента в HTML?
- а) `background-color: red;`

- б) color: red;
- в) bg-color: red;
- г) font-color: red;

12. Як вставити коментар у код HTML?

- а) `<!-- Comment -->`
- б) `// Comment`
- в) `* Comment *`
- г) `# Comment`

Ключ до тесту

- 1. а) Hyper Text Markup Language
- 2. в) `<title>`
- 3. а) `<a>`
- 4. а) ``
- 5. в) ``
- 6. в) `<h1>`
- 7. в) ``
- 8. а) `<i>`
- 9. б) `<hr>`
- 10. а) `<table>`
- 11. а) background-color: red;
- 12. а) `<!-- Comment -->`

Тест до практичної роботи № 4

1. Який елемент HTML використовується для вставки зображення на веб-сторінку?

- а) ``
- б) `<picture>`
- в) `<image>`
- г) `<src>`

2. Яким атрибутом вказується шлях до зображення в тезі ``?

- а) src
- б) alt
- в) title
- г) href

3. Який елемент HTML використовується для відтворення

аудіо на веб-сторінці?

- а) <audio>
- б) <sound>
- в) <music>
- г) <play>

4. Яким атрибутом вказується шлях до аудіофайлу в елементі <audio>?

- а) src
- б) alt
- в) title
- г) href

5. Який елемент HTML використовується для відтворення відео на веб-сторінці?

- а) <video>
- б) <movie>
- в) <play>
- г) <media>

6. Яким атрибутом вказується шлях до відеофайлу в елементі <video>?

- а) src
- б) alt
- в) title
- г) href

7. Який CSS властивість використовується для задання фонового зображення елементу?

- а) background-image
- б) background-color
- в) background-style
- г) image-source

8. Яким CSS властивостям можна задати анімацію елемента?

- а) animation-name, animation-duration, animation-delay
- б) animate, duration, delay
- в) transition-name, transition-duration, transition-delay
- г) keyframes, duration, delay

9. Який формат зображень найчастіше використовується для веб-сторінок з малою втратою якості?

- а) JPEG

- б) GIF
- в) PNG
- г) SVG

10. Який формат відео зазвичай має найменший розмір файлу при збереженні з високою якістю відео?

- а) MP4
- б) AVI
- в) MKV
- г) MOV

Ключ до тесту

- 1. а)
- 2. а) src
- 3. а) <audio>
- 4. а) src
- 5. а) <video>
- 6. а) src
- 7. а) background-image
- 8. а) animation-name, animation-duration, animation-delay
- 9. а) JPEG
- 10.а) MP4

Тест до практичної роботи № 5

- 1. Що таке HTML форма?
 - а) Елемент HTML, який використовується для створення структури веб-сторінки
 - б) Елемент HTML, який використовується для відображення текстового поля
 - в) Елемент HTML, який використовується для введення даних користувачем
- 2. Який елемент HTML використовується для створення форми?
 - а) <form>
 - б) <input>
 - в) <label>
- 3. Який атрибут використовується для вказання методу передачі даних з форми на сервер?

- a) action
 - б) method
 - в) target
4. Який метод передачі даних з форми на сервер використовується для передачі даних відкритим текстом?
- a) GET
 - б) POST
 - в) PUT
5. Який елемент HTML використовується для створення кнопки в формі?
- a) <button>
 - б) <select>
 - в) <textarea>
6. Який елемент HTML використовується для створення випадаючого списку в формі?
- a) <button>
 - б) <select>
 - в) <textarea>
7. Яким атрибутом вказується тип поля вводу (наприклад, текстове поле чи пароль)?
- a) type
 - б) name
 - в) value
8. Який елемент HTML використовується для створення прапорця в формі?
- a) <input type="text">
 - б) <input type="checkbox">
 - в) <input type="radio">
9. Який елемент HTML використовується для створення радіокнопок в формі?
- a) <input type="text">
 - б) <input type="checkbox">
 - в) <input type="radio">
10. Який елемент HTML використовується для створення текстової області вводу в формі?
- a) <input type="text">
 - б) <input type="checkbox">

в) <textarea>

11. Як можна перевірити правильність введених даних в формі перед їх відправкою на сервер?

- а) Використовуючи JavaScript
- б) Використовуючи CSS
- в) Використовуючи HTML

Ключ до тесту

- 1. в) Елемент HTML, який використовується для введення даних користувачем
- 2. а) <form>
- 3. б) method
- 4. а) GET
- 5. а) <button>
- 6. б) <select>
- 7. а) type
- 8. б) <input type="checkbox">
- 9. в) <input type="radio">
- 10. в) <textarea>
- 11. а) Використовуючи JavaScript

Підсумковий тест з вибіркового модуля

- 1. Структура веб-сайтів включає обов'язково
 - а) меню сайта
 - б) шапка сайта
 - в) гіперпосилання
 - г) авторське право
 - д) головна сторінка
- 2. Яка структура веб-сайта найдоцільніша у використати для навчального матеріалу?
 - а) Деревоподібна
 - б) Павутина
 - в) Лінійна
 - г) Комбінована
- 3. Яка структура забезпечує таку структуру сайту: усі сторінки Web-сайту містять посилання на інші сторінки, користувач може легко перейти з будь-якої сторінки практично на будь-яку іншу.
 - а) Деревоподібна
 - б) Павутина

- в) Лінійна
 - г) Комбінована
4. Такої системи навігації не існує
- а) Система відгуків
 - б) Використання навігаційних карт
 - в) За допомогою кнопок
 - г) Текстова система
5. Головна сторінка сайта, має відповідати таким вимогам
- а) відображати тематику сайта даними, що постійно оновлюються
 - б) містити розділ з актуальними
 - в) містити відомості про спосіб зворотного зв'язку
 - г) бути цікавою
 - д) містити інструмент реалізації пошуку по сайту
6. Який з варіантів не є доменом?
- а) .com
 - б) .www
 - в) .org
 - г) .ua
7. Об'єднання тематично пов'язаних веб-сторінок і документів під одним доменним іменем – це
- а) Сайт
 - б) Гіпертекст
 - в) Хостинг
 - г) Контент
8. Зміст – це
- а) Сайт
 - б) Гіпертекст
 - в) Хостинг
 - г) Контент
9. Етапи створення сайта, які зазвичай передують його наповненню контентом
- а) Обслуговування сайту
 - б) Просування сайту
 - в) Планування структури сайту
 - г) Створення дизайну
10. Оберіть основні елементи веб-сторінки

- а) контент
- б) навігаційні кнопки
- в) форма
- г) елементи дизайну

11. Елементи роботи з інформацією

- а) зберігання
- б) пошук
- в) опрацювання
- г) аналіз

Ключ до тесту

1. а) меню сайта
2. в) Лінійна
3. б) Павутина
4. а) Система відгуків
5. а) відображати тематику сайта
6. б) .www
7. а) Сайт
8. г) Контент
9. в) Планування структури сайту
10. а) контент
11. всі відповіді правильні:
а) зберігання, б) пошук, в) опрацювання, г) аналіз.

**Вибірковий модуль «Креативне програмування»
(35 годин)**

Укладач Я.В.Левицький

Календарно-тематичний план

№ Уроку	Тема уроку	Дата
Цифрове мистецтво та творчість		
1	Цифрове мистецтво. Дизайн та код. Генеративне мистецтво.	
2	Програмування як середовище для творчості.	
3	Класифікація мов програмування.	
4	Особливості середовища розробки.	
5	Структура програмного проекту	
6	Основні елементи мови програмування. Використання змінних і виразів.	
7	Реалізація алгоритмів з розгалуженням	
8	Реалізація циклічних алгоритмів.	
Графічні побудови та взаємодії		
9	Полотно. Пікселі. Координати. Кольори.	
10	Основні форми: точка, лінія, прямокутник, еліпс	
11	Обробка подій. Інтерактивність.	
12	Правила написання читабельного коду. Коментарі у тексті програми.	
Функції		
13	Модульність.	
14	Функції. Бібліотеки та модулі.	
15	Передавання значень у функцію та з неї. Формальні та фактичні параметри.	
16	Рекурсія. Рекурсивні побудови.	
17	Фрактали як самоподібні структури.	

№ Уроку	Тема уроку	Дата
Об'єкти та класи		
18	Поняття об'єкта, властивості об'єкта, класу як об'єктного типу даних	
19	Поняття події та методу.	
20	Створення програмних об'єктів.	
21	Опис класів та створення об'єктів.	
22	Взаємодія об'єктів.	
23	Поняття, реалізація та застосування масивів (списків).	
24	Функції та методи опрацювання масивів (списків)	
25	Текстові рядки як масиви символів	
26	Зображення як цілісний об'єкт та як масив пікселів	
Мультимедіа		
27	Відео як цілісний об'єкт та як масив зображень. Анімації	
28	Бібліотеки для роботи з мультимедійними даними	
Інтерфейс програмного продукту		
29	Графічний інтерфейс користувача.	
30	Прикладний програмний інтерфейс.	
31	Дані сенсорів та датчиків.	
32	Зовнішні джерела даних.	
33	Поняття та приклади інтерактивних інсталяцій.	
34	Виконання навчальних проєктів.	
35	Представлення та захист навчальних проєктів.	

Література для вчителя

1. Васильєв О.М. Програмування мовою Python. Тернопіль: Навчальна книга – Богдан, 2019. 504 с.
2. Маттес Е. Пришвидшений курс PYTHON. Практичний, проектно-орієнтований вступ до програмування. Львів: Видавництво Старого Лева, 2021. 600 с.
3. Руденко В.Д. Інформатика: креативне програмування (модуль для учнів 10-11 класів, рівень стандарту). Харків: «Ранок». 2020. 160 с.
4. Електронний підручник Python <https://docs.python.org/uk/3/tutorial/index.html>
5. Електронний підручник Python <https://www.w3schools.com/python/>
6. Збірник задач та розв'язань із програмування / С.М. Бондаренко, В.В. Зуб, О.І. Коваленко та ін. ; за заг. ред. О. Є. Баранової, Ю. М. Літоша. Чернігів: ЧОППО імені К. Д. Ушинського, 2016. Ч.1. 40 с. URL: http://choippo.cn.sch.in.ua/Files/downloadcenter/%D0%A0%D0%BE%D0%B7%D0%B2%D1%8F%D0%B7%D0%BA%D0%B8%2010-12.2015_%D0%BE%D1%81%D1%82%D0%B0%D0%BD_0.pdf
7. Збірник задач та розв'язків із програмування / Н. П. Хрол, В. М. Харченко, Л. Б. Коваленко та ін.; за заг. ред. Ю. М. Літоша, О. Є. Баранової, О. М. Смірної. Чернігів: ЧОППО імені К. Д. Ушинського, 2016. Ч.2. 71 с. URL: <http://choippo.cn.sch.in.ua/Files/downloadcenter/%D0%97%D0%91%D0%86%D0%A0%D0%9D%D0%98%D0%9A%20%D1%87%202%D0%BE%D1%81%D1%82.pdf>
8. Білоусова Л.І. Краса простих задач або до питання про використання мов програмування у навчанні школярів інформатики // Комп'ютер у школі та сім'ї. – 2014. – №1. – С.18-22.
9. Бондаренко С.М. Збірник задач та розв'язків II етапу Всеукраїнської учнівської олімпіади з інформатики 2014-

2015 навчального року / С.М. Бондаренко, В.В. Зуб, Ю.М. Літош. – Чернігів: ЧОШПО імені К.Д. Ушинського, 2015. 35 с.

10.Крєневич А.П. Python у прикладах і задачах. Частина 1. Структурне програмування Навчальний посібник із дисципліни "Інформатика та програмування" – К.: ВПЦ "Київський Університет", 2017. – 206 с. URL: <http://www.matfiz.univ.kiev.ua/uploads/books/python2017.pdf>

11.Крєневич А.П. Python у прикладах і задачах. Частина 2. Об'єктно-орієнтоване програмування. Навчальний посібник. – К.: ВПЦ "Київський Університет", 2020. 152 с. URL: <http://www.mechmat.univ.kiev.ua/wp-content/uploads/2021/01/python-u-prykladakh-i-zadachakh.-ch2-oop.pdf>

Література для учнів

1. Васильєв О.М. Програмування мовою Python. Тернопіль: Навчальна книга – Богдан, 2019. 504 с.
2. Маттес Е. Пришвидшений курс PYTHON. Практичний, проєктно-орієнтований вступ до програмування. Львів: Видавництво Старого Лева, 2021. 600 с.
3. Руденко В.Д. Інформатика: креативне програмування (модуль для учнів 10-11 класів, рівень стандарту). Харків: «Ранок». 2020. 160 с.

Програмне забезпечення:

1. IDLE (Integrated Development and Learning Environment) – редактор за замовченням, який супроводжує Python. Підходить для розробників-початківців. Інструмент IDLE можна використовувати в Mac OS, Windows і Linux безкоштовно.

Функціонал IDLE включає можливість пошуку декількох файлів, інтерактивний інтерпретатор із підсвічуванням синтаксису та повідомленнями про помилки та введення/виведення, дуже потужний налагоджувач тощо.

2. PyCharm – широко використовувана IDE Python, створена компанією JetBrains. Підходить для професійних розробників і полегшує розробку великих проектів Python.

Ціна: Freemium

Функціонал PyCharm включає підтримку JavaScript, CSS і TypeScript, «розумну» навігацію по коду, швидкий і безпечний рефакторинг коду, доступ до баз даних безпосередньо з IDE та інше.

3. Visual Studio Code – безкоштовне IDE з відкритим кодом, створене Microsoft.

4. Sublime Text 3 – популярний багатомовний безкоштовний редактор коду Він легко налаштовується, а також забезпечує високу швидкість розробки та надійність.

Підтримує підсвічування синтаксису, спеціальні команди користувача для використання IDE, ефективне управління каталогом проектів, додаткові пакети для веб-розробки та наукової розробки Python.

5. Atom – безкоштовний редактор коду з відкритим кодом від GitHub, який підтримує розробку на Python. Atom схожий на Sublime Text і надає майже ті самі функції, акцентуючи увагу на швидкості та зручності використання

Особливості Atom:

- Підтримка великої кількості плагінів;
- Розумне автозаповнення;
- Підтримує спеціальні команди для взаємодії користувача з редактором;
- Підтримка кросплатформенної розробки.

6. Jupyter – широко використовується в галузі науки про дані. Це безкоштовне, просте у використанні, інтерактивне середовище, яке дозволяє обмінюватися кодом і візуалізувати його.

Особливості Jupyter:

- Підтримка числових обчислень і робочого процесу машинного навчання;

- Комбінування коду, тексту і зображення для кращої взаємодії з користувачем;
- Взаємодія наукових бібліотек даних, таких як NumPy, Pandas і Matplotlib.

7. Spyder – безкоштовне IDE з відкритим вихідним кодом, яке найчастіше використовується для наукових розробок. Постачається з дистрибутивом Anaconda, популярним для науки про дані та машинного навчання

Особливості Spyder:

- Підтримка автоматичного доповнення та розбиття коду;
- Підтримує побудову різних типів діаграм і маніпулювання даними;
- Інтеграція наукових бібліотек даних, таких як NumPy, Pandas і Matplotlib.

8. PyDev – безкоштовний потужний інтерпретатор Python, який поширюється як сторонній плагін для Eclipse IDE

Особливості PyDev:

- Інтеграція Django, автоматичне доповнення коду та покриття коду;
- Підтримує підказку типу, рефакторинг, а також налагодження та аналіз коду;
- Гарна підтримка веб-розробки Python.

9. Thonny — ідеальне безкоштовне середовище розробки для навчання програмуванню на Python

Thonny має:

- Простий налагоджувач;
- Оцінку функції;
- Автоматичне виявлення синтаксичних помилок;
- Детальний перегляд змінних, які використовуються в програмі або проекті Python.

Практичні завдання до уроків:

Урок 1. Цифрове мистецтво. Дизайн та код.

Генеративне мистецтво.

Створіть презентацію “Генеративне мистецтво”.

Урок 2. Програмування як середовище для творчості.

Створіть презентацію «Рейтинг мов програмування».

Урок 3. Класифікація мов програмування.

Дослідіть онлайніві середовища, які підтримують мову програмування Python. Визначте їхні переваги та недоліки. Зробіть висновок. Результати дослідження оформіть в текстовому документі.

Урок 4. Особливості середовища розробки.

Створіть діалогову програму «Знайомство з учнем. Підпис зошита».

Програма має містити питання, користувач повинен давати відповіді. Попередньо потрібно запитати у користувача його ім'я та прізвище, клас, заклад освіти, та предмет, з якого підписати зошит.

Вивести інформацію у вигляді підписаного зошита з інформатики.

Урок 5. Структура програмного проєкту.

Програма "Куб". Дано довжину ребра куба. Знайти його об'єм та площу всієї поверхні.

Урок 6. Основні елементи мови програмування.

Використання змінних і виразів.

Створити програму «Кафе». В програмі передбачте можливість введення кількості придбаних одиниць товарів, розрахунок вартості покупки, введення суми, внесеної покупцем та решти, що повертається покупцеві.

Програма «Піца». В програмі передбачте можливість вибору складових піци, розрахунок вартості покупки,

введення суми, внесеної покупцем та решти, що повертається покупцеві.

Урок 7. Реалізація алгоритмів з розгалуженням.

1. Оклад працівника або працівниці дорівнює s грн. За якісне й дострокове виконання завдання нараховується премія в розмірі 50% окладу. Розробіть програму визначення реальної заробітної платні.
2. Дано два дійсних числа. Розробіть програму, за допомогою якої менше число замінюється нулем, а більше – одиницею.
3. Дано ціле число n . Розробіть програму, за допомогою якої це число збільшується на 7, якщо воно більше 15, і зменшується на 5, якщо воно менше або дорівнює 15.
4. Тренерка формує команду для гри в баскетбол з учениць, які на зріст не нижчі ніж 180 см. Розробіть програму визначення, чи потрапить у цю команду дівчина зростом h см?
5. Дано рівносторонній трикутник зі стороною b . Розробіть програму визначення, чи можна в трикутник вписати коло з радіусом r .
6. Відомі три найкращі результати учнівства школи в стрибках у довжину. Розробіть програму, за допомогою якої за результатами стрибків визначаються прізвиська учнів.
7. У п'ятницю в 9 класі такі уроки: 1 – історія, 2 – математика, 3 – географія, 4 – інформатика, 5 – фізкультура. Розробіть програму, за допомогою якої визначається назва предмета за його номером у розкладі.
8. Знайдіть в Інтернеті та складіть список із п'яти найбільших міст України. Розробіть програму, за допомогою якої визначається назва міста за його номером у цьому списку.
9. Дано три сторони трикутника. Розробіть програму, за допомогою якої визначається, чи є цей трикутник прямокутним.

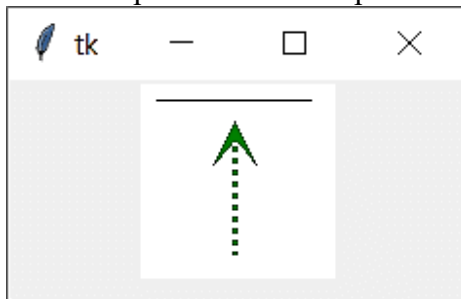
Урок 8. Реалізація циклічних алгоритмів.

1. Розробіть код обчислення суми для чисел: 2, 7, 21, 9, 33, 13.
2. Розробіть код, обчислення суми непарних чисел, що більші 7, але менші 25.
3. Розробіть код обчислення суми чисел натурального ряду, максимальне значення якого не перевищує 7.
4. Перший член геометричної прогресії 6, а її знаменник – 0.5. Розробіть код обчислення значень членів прогресії, більших 0.6, і визначення номера останнього члена прогресії, що підсумовується.
5. Дано куб, сторони якого набувають п'ять значень: 3, 4.5, 6, 7.5, 9. Розробіть код визначення об'єму для кожного з них.
6. Радіус першої кулі дорівнює 2 см, а радіус кожної наступної збільшується на 0.5 см. Розробіть код для визначення бокових поверхонь перших шести куль.
7. Відомо результати плавання вільним стилем на дистанції 50 м п'яти учнів кожного з трьох 10 класів школи. Розробіть код, за допомогою якого визначається середній час запливу учнями кожного класу.
8. У банк покладено S грн під N відсотків річних. Розробіть код, за допомогою якого визначається кількість років, через які сума вкладу буде не менше N грн

Урок 9. Полотно. Пікселі. Координати. Кольори.

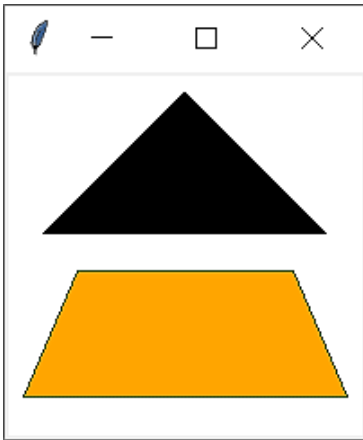
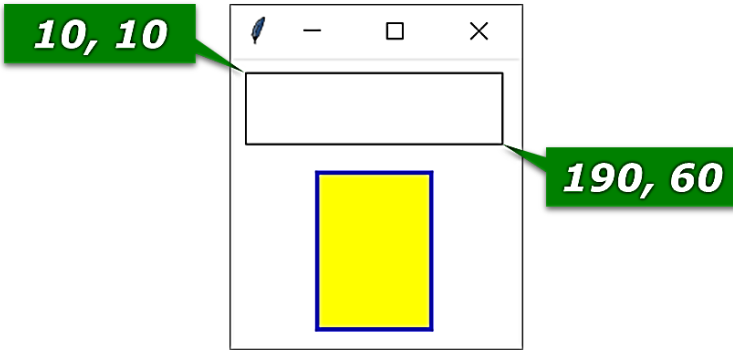
Вправа 1. Стрілка.

Завдання. Створіть малюнок стрілки за зразком



Вправа 2. Прямокутники.

Завдання. Створіть малюнок двох прямокутників за зразком:

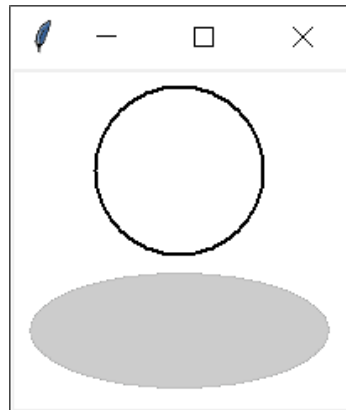


Вправа 3.
Багатокутники.

Завдання. Створіть малюнок геометричних фігур за зразком.

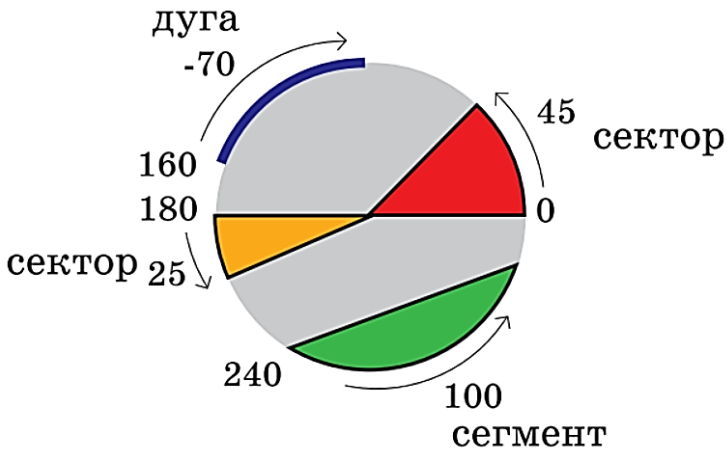
Вправа 4. Тінь.
Завдання. Створіть малюнок за зразком:

Врахуйте, що овал – це фігура, вписана в уявний прямокутник, що визначається діагоналлю з координатами: ліва верхня точка (50, 10), нижня права – (150, 100).



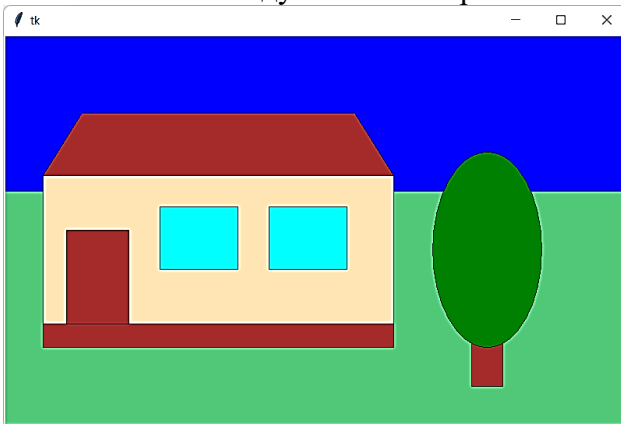
Вправа 5. Елементи круга.

Завдання. Використовуючи модуль Canvas, створіть зображення за зразком:



Урок 10. Основні форми: точка, лінія, прямокутник, еліпс.

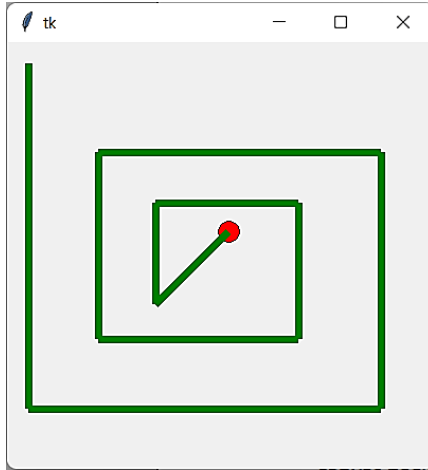
Завдання. Створіть в середовищі *Python IDLE* зображення за власним задумом або за зразком.



Урок 11. Обробка подій. Інтерактивність.

Завдання. Створити програму, в якій реалізовано рух кульки по полотну. Об'єкт має змінювати положення у

відповідь на натискання клавіш управління курсором, залишаючи слід на полотні.



Урок 12. Правила написання читабельного коду. Коментарі у тексті програми.

1. Відкрийте вікно середовища **IDLE**.
2. Створіть нове вікно для введення коду. Для цього виконайте **File** ⇒ **New file**.
3. Уведіть у новому вікні команди для створення вікна виконання проєкту:
 - а) команду **from tkinter import *** для підключення до проєкту модуля створення графічного інтерфейсу;
 - б) команду **root = Tk()** для створення вікна проєкту.
4. Уведіть команду **root.bind('<1>', click)** для зв'язування з вікном проєкту процедури з іменем **click** для опрацювання події **Click**.
5. Створіть обробник події з іменем **click**. Для цього:
 - Установіть курсор після останнього символу першого рядка проєкту та натисніть клавішу **Enter**.
 - Уведіть заголовок обробника події **def click(event):** та натисніть клавішу **Enter**. Зверніть увагу на автоматичне розміщення курсора з відступом від лівої межі вікна редактора тексту.

- Уведіть після заголовка процедури команду **root.geometry('500x300')**, яка встановить ширину вікна 500 пікселів, висоту – 300 пікселів.
 - Уведіть команду **root.title('Я змінюю значення властивостей вікна')**, яка виведе в рядку заголовка вікна текст «Я змінюю значення властивостей вікна».
 - Уведіть команду **root['bg'] = 'yellow'**, яка встановить жовтий колір фону вікна.
 - Збережіть проєкт у вашій папці у файлі з іменем **Завдання 12**.
 - Запустіть проєкт на виконання.
 - Установіть вказівник у вікні, що відкрилося, та клацніть ліву кнопку миші. Зверніть увагу, як змінився колір фону вікна, його ширина та текст у рядку заголовка.
 - Закрийте вікно виконання проєкту.
6. Додайте до проєкту команди збільшення висоти вікна проєкту на 150 пікселів після клацання лівої кнопки миші. Для цього в кінець тексту процедури:
 - Вставте команду $h = root.wininfo_height() + 150$, яка обчислить значення висоти вікна, більше на 150 пікселів від поточного значення.
 - Встановіть для вікна обчислене значення висоти командою **root.geometry('500x{h}'.format(h))**.
 7. Збережіть змінену версію проєкту і запустіть його на виконання.
 8. Установіть вказівник у вікні, що відкрилося, і клацніть кілька разів ліву кнопку миші. Зверніть увагу, як змінюється висота вікна.
 9. Додайте до проєкту команди відкриття вікна повідомлень після клацання лівої кнопки миші. Для цього:
 - Вставте після першого рядка проєкту команду **from tkinter.messagebox import ***

для використання команд відкриття вікон повідомлень.

- Вставте в кінець тексту процедури команду *showinfo('Події', 'Україна – єдина країна!')*, яка відкриє вікно повідомлень з текстом у рядку заголовка Події та текстом повідомлення «Україна – єдина країна!»

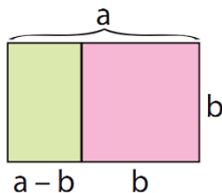
10. Збережіть змінену версію проєкту і запусіть його на виконання.
11. Установіть вказівник у вікні, що відкрилося, і клацніть ліву кнопку миші.
12. Зверніть увагу на вміст вікна повідомлень.

Урок 13. Модульність.

Завдання. Прямокутник, довжини сторін a і b якого задовольняють умові:

$$\frac{a}{b} = \frac{b}{a - b}$$

називають «золотим».



Визначте за відомими довжинами сторін, чи є прямокутник «золотим».

1. Проаналізуйте постановку задачі та складіть математичну модель задачі.

Що дано? a, b – довжини сторін.

Що треба знайти? Відповідь Так або Ні.

Який зв'язок між величинами?

Якщо $\frac{a}{b} = \frac{b}{a-b}$, то вивести Так,
інакше – вивести Ні.

Які є обмеження на дані? $0 < b < a$.

2. У **Python IDLE** виберіть команду **File** \Rightarrow **New File**.
3. Запишіть оператори введення значень змінних a і b :


```
a = float(input('a = ?'))
b = float(input('b = ?'))
```
4. Запишіть оператор розгалуження для перевірки відповідності значень змінних


```
if abs(a/b-b/(a-b))<0.001:
    print('Так')
else:
    print('Ні')
```

Примітка. Через особливості подання в пам'яті дробових чисел порівняння двох виразів за допомогою операції « $==$ » не завжди дає очікуваний результат. Вважатимемо значення рівними, якщо вони відрізняються не більше ніж на деяку досить малу величину (точність). Тут 0,001. Функція `abs` повертає модуль аргументе.
5. Збережіть файл з іменем **Завдання 13**. Запустіть програму, проаналізуйте результат її виконання у вікні консолі. Випробуйте програму для різних значень змінних a і b :
 - а) $a = 8.09, b = 5$;
 - б) $a = 10, b = 5$.

Урок 14. Функції. Бібліотеки та модулі.

Вправа 1. Подарункова коробка.

Завдання. З комп'ютерного диска (діаметр 120 мм) восьмикласники виготовили квадратну заготовку для сувеніру. Складіть програму, яка визначатиме сторону подарункової коробки для цього сувеніру.

Вправа 2. Кількість теплоти.

Завдання. Складіть програму для знаходження кількості теплоти, яка потрібна для нагрівання тіла масою m , з початковою і кінцевою температурами тіла t_1 , і t_2 ,

питомою теплоємністю речовини c , яка обчислюється за формулою:

$$Q = cm(t_2 - t_1)$$

Для обчислення використайте такі дані:

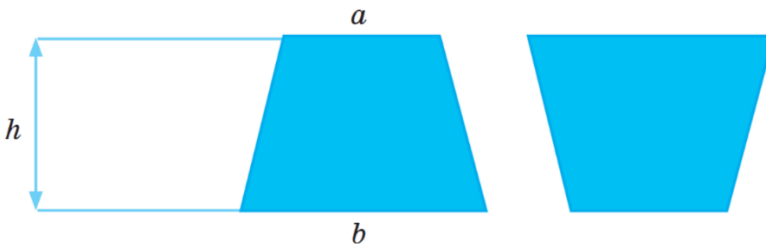
$m = 5$ кг, $t_1 = 20$ °С; $t_2 = 100$ °С, $c = 2000$ Дж/°С.

Якого типу буде змінна Q ?

Вправа 3. Доріжка в парку.

Завдання. Доріжку у парку вирішили встелити плиткою, яка має форму трапеції. Складіть програму, яка визначає найменшу кількість плиток для заданої площі P , якщо відомо, що ширина доріжки кратна висоті h трапеції. Врахуйте, що площу трапеції знаходять за формулою:

$$S = \frac{(a + b)}{2} h$$



Вправа 4. Координатна площина.

Завдання. Відомо, що відстань між двома точками, які задані на координатній площині своїми координатами (x_1, y_1) та (x_2, y_2) , визначається за формулою:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Складіть програму, за допомогою якої знаходять:

- 1) периметр трикутника, заданого координатами своїх трьох вершин;
- 2) площу прямокутника, побудованого на сторонах AB та AC , для яких задано координати вершин A, B, C .

Урок 15. Передавання значень у функцію та з неї. Формальні та фактичні параметри.

Завдання. Скласти програму для розв'язування задачі.

Задача. На стадіоні місця класу А коштують a грн, місця класу В – b грн, а місця класу С – c грн. Обчислити загальну суму прибутку від продажу квитків.

1. У **Python IDLE** виберіть команду **File** \Rightarrow **New File**. Збережіть файл з іменем **Завдання 15**.
2. Запишіть оператори для введення значень змінних a , b , c .
3. Опишіть функцію $kvitok(a, b, c)$, яка запитує, скільки продано квитків класу a , і обчислює суму отриманого прибутку від продажу квитків цього класу.

```
def kvitok(a, b, c):
```

```
    k_a = int(input('Скільки продано квитків класу А? '))
```

```
    sum_a = a*k_a
```

```
    return sum_a
```

4. Додайте до опису функції $kvitok(a, b, c)$ оператори для введення кількості проданих квитків класів В і С й обчислення суми, отриманої від продажу квитків кожного класу окремо. Після цього оператор `return` слід змінити так, щоб функція повертала три значення:

```
return sum_a, sum_b, sum_c
```

5. В основній програмі обчисліть загальну суму прибутку від продажу квитків:

```
s_a, s_b, s_c = kvitok(a, b, c)
```

```
suma = s_a+s_b+s_c
```

6. Запишіть оператор виведення значення загальної суми. Випробуйте роботу програми для різних початкових значень.

Урок 16. Рекурсія. Рекурсивні побудови.

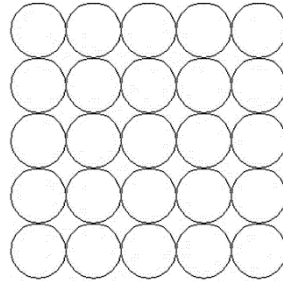
1. Розробіть програму з рекурсивною функцією обчислення n перших членів геометричної прогресії, перший член якої дорівнює a і знаменник q .
2. Розробіть програму з використанням рекурсивної функції обчислення кількості цифр у десятковому натуральному числі a .

Урок 17. Фрактали як самоподібні структури.

Вправа 1. Створити програму розв'язання задачі в середовищі Python.

Задача. Створити програму побудови зображення згідно зразка. Радіус кола = 30

Збережіть файл програми з іменем **1.Коло 5x5**.

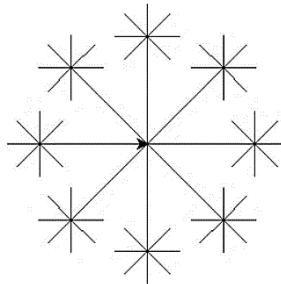


Вправа 2. Створити програму розв'язання задачі в середовищі Python.

Задача. Створити програму побудови зображення сніжинки.

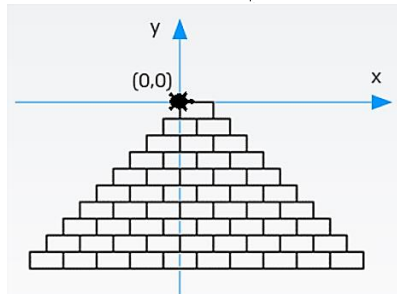
Довжина променю великої сніжинки дорівнює 100, маленької сніжинки – 30.

Збережіть файл програми з іменем **2.Сніжинка**.



Вправа 3. Створити програму розв'язання задачі в середовищі Python.

Задача. Створити програму побудови піраміди з цеглин. Розмір цеглинки – 20 x 40.



Збережіть файл програми з іменем *3.Пірамідка*.

Урок 18. Поняття об'єкта, властивості об'єкта, класу як об'єктного типу даних.

Завдання. Створити об'єкт класу Тварини.

1. У **Python IDLE** виберіть команду **File** ⇒ **New File**.
2. Створіть клас з іменем **Animal()**. Метод `__init__()` повинен містити атрибути, що характеризують вид тварини (**kind**), зріст (**height**), кількість лап (**legs**).

```
class Animal():
```

```
    def __init__(self, kind, height, legs):  
        self.kind = kind  
        self.height = height  
        self.legs = legs
```

3. Уведіть із клавіатури значення характеристик тварини і збережіть значення у змінних **k**, **h**, **l**:

```
k = input('Вид тварини')  
h = int(input('Зріст'))  
l = int(input('Кількість лап'))
```

4. Створіть екземпляр класу **Animal** з іменем **a1**, атрибути якого набувають значення змінних **k**, **h**, **l**:

```
a1 = Animal(k, h, l)
```

5. Виведіть у консоль характеристики об'єкта **a1**:

```
print('Вид – ', a1.kind, 'Зріст – ', , a1.height, 'Лап – ', a1.legs)
```

6. Збережіть файл із назвою **Завдання 18**.

7. Запустіть програму. Уведіть значення для змінних **k**, **h**, **l**, які відповідають об'єкту мавпа.

Урок 19. Поняття події та методу.

Завдання. Створити об'єкт класу Тварини.

1. У **Python IDLE** виберіть команду **File** ⇒ **New File**.
2. Створіть клас **Animal()** з атрибутами **kind** (вид Тварини) і **legs** (кількість лап).

```
class Animal():
```



```
def __init__(self, kind, legs):
    self.kind = kind
    self.legs = legs
```

3. Додайте до класу метод **description()**, який виводить повідомлення про кількість лап в об'єкта – екземпляра класу:

```
def description(self):
    if self.legs == 1: print(self.kind, 'має', self.legs,
                              'кінцівку')
    else: print(self.kind, 'має', self.legs, 'кінцівки')
```

4. Створіть екземпляр класу **Animal()**, який представляє об'єкт слон:

```
a1 = Animal('слон', 4)
```

5. Запишіть виклик методу **description()** для об'єкта **a1**:

```
a1.description()
```

6. Створіть екземпляри класу **Animal()**, які представляють об'єкти курка, устриця. Запишіть виклики методу **description()** для цих об'єктів.

7. Збережіть файл із назвою **Завдання 19**. Запустіть програму на виконання. Проаналізуйте результати у вікні консолі.

Урок 20. Створення програмних об'єктів.

Завдання: скласти програму **Подорож**.

Хід роботи

У **Python IDLE** виберіть команду **File** ⇒ **New File**.

1. Створіть клас з іменем **Car()**.

```
class Car():
    def __init__(self, mileage, fuel):
        self.mileage = mileage      # Пройдений шлях
        self.fuel = fuel            # Кількість пального
```

2. Додайте до класу метод **vitr(self)**, який зменшує значення атрибута **fuel**:

```
def vitr(self):      # Витрати пального на 50 км
```

```
self.fuel = self.fuel-3
    print('залишилось пального'+str(car1.fuel))
```

3. Додайте до класу метод **zapr(self)** (заправка), який збільшує значення атрибута **fuel** на 10 л.

```
def zapr(self ):                # Заправка
    self.fuel = self.fuel+10
    print('залишилось пального', car1.fuel)
```
4. Утворіть екземпляр **car1** класу **Car()** із параметрами 0, 0.

```
car1 = Car(0, 0)
```
5. Виведіть значення пробігу створеного об'єкта.

```
print ('Пробіг', car1.mileage)
```
6. «Заправте» автомобіль:

```
car1.zapr().
```
7. Запишіть оператор для введення відстані, яку потрібно подолати, і збереження значення у змінній **a**:

```
a = int(input('відстань?'))
```
8. Поки відстань **a > 0**, потрібно повторювати дії: надрукувати **ЇДЕМО!**; зменшити значення відстані на 50 км; викликати для об'єкта **car1** метод **vitr()**; якщо пального залишається менше ніж 3 л, викликати для об'єкта **car1** метод **zapr()**.
Запишіть оператор циклу **while**, що реалізує алгоритм руху автомобіля:

```
while (a>0):
    print ('ЇДЕМО!')
    a = a-50
    car1.vitr()
    if (car1.fuel < 3): car1.zapr()
```
9. Коли шлях пройдено, повідомте про це:

```
print('ФІНІШ!')
```
10. Збережіть файл у власні папці папці з іменем **Завдання 20**.
11. Виконайте програму для різних значень відстані, яку потрібно проїхати.

Урок 21. Опис класів та створення об'єктів.

Завдання. Запрограмувати рух об'єктів класу Ball (М'яч) по вертикалі з відбиттям від меж вікна.

1. У **Python IDLE** виберіть команду **File** ⇒ **New File**.

2. Імпортуйте необхідні модулі:

```
from tkinter import*
from random import*
from time import*
```

3. Створіть вікно програми і додайте полотно розміром 500 x 400.

```
tk = Tk()
canvas = Canvas(tk, width = 500, height = 400)
canvas.pack()
```

4. Створіть клас **Ball()** з атрибутами *x*, *y* (координати правого верхнього кута квадрата, в який уписано круг) і *color* (колір кульки).

```
class Ball:
```

```
    def __init__(self, canvas, x, y, color):
        self.canvas = canvas
        self.id = canvas.create_oval(x, y, x+50, y+50, fill = color)
        self.dy = 2 # Величина зміни координати y
```

5. Додайте до опису класу Ball метод» в якому після перевірки, чи дійшов об'єкт до межі вікна, і змінення за потреби напрямку руху, виконуватиметься переміщення в нове положення.

```
    def ruh(self):
        pos = canvas.coords(self.id)
        if pos[1] > 348 or pos[1] < 2: self.dy = -1*self.dy
        canvas.move(self.id, 0, self.dy)
```

6. Створіть список із 10 об'єктів класу Ball із випадковими координатами:

```
list_ball = []
colors = ['red', 'orange', 'yellow', 'green', 'blue']
for i in range(10):
```

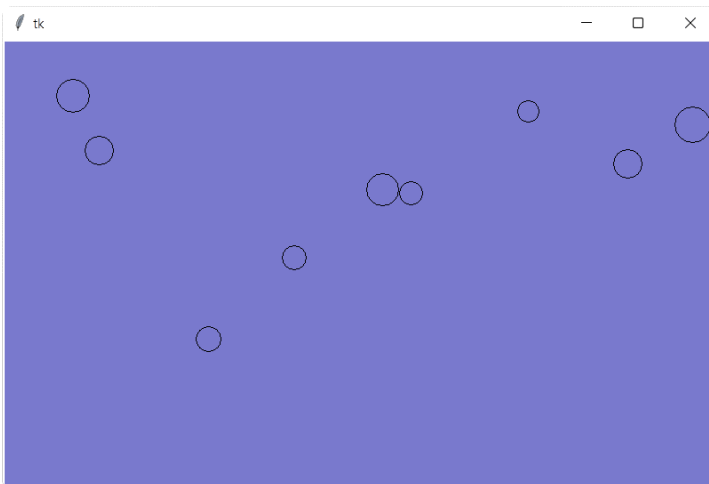
```
x = randint(10, 400)
y = randint(10, 350)
list_ball.append(Ball(canvas, x, y, colors[i%5]))
```

7. Організуйте повторення виклику методу `ruh()` для кожного елемента списку `list_ball`.

```
def play():
    for i in range(len(list_ball)):
        list_ball[i].ruh()
    tk.update()      # Оновлення полотна
    tk.after(10, play) # Затримка виконання програми
play()
tk.mainloop()
```

Урок 22. Взаємодія об'єктів.

Скласти ігрову програму *Ловець бульбашок*. На ігровому полотні через певні проміжки часу з'являються бульбашки. Гравець ловить бульбашки, клацаючи їх. Якщо кількість бульбашок перевищує допустиму, гра припиняється.



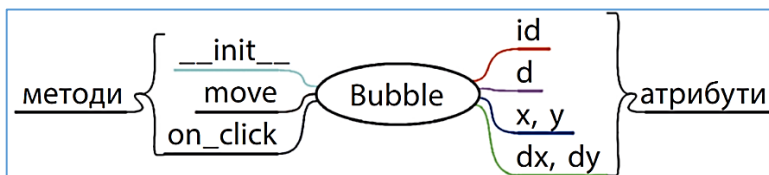
У Python IDLE виберіть команду **File** ⇒ **New File**.

1. Завантажте модулі, що потрібні для реалізації руху об'єктів. Створіть вікно програми і додайте полотно розміром 800 x 500:

```
from tkinter import*
from random import*
from time import*
tk = Tk()
canvas = Canvas(tk, width = 800, height = 500, bg='blue')
canvas.pack()
```

2. Створіть змінні для керування режимом гри:
maxV=10 # За такої кількості бульбашок на полотні гра зупиниться
genT=1000 # Проміжок часу між появою бульбашок

Створіть клас **Bub()** – модель бульбашки:



Екземпляр цього класу повинен:

- створюватись на полотні **canvas** вікна **tk**;
- мати випадкові координати **x** і **y** (у межах полотна);
- зразу ж після створення починати рух, зміщуючись на кожному кроці на **dx** і **dy** пікселів уздовж відповідних осей;
- відбиватись від країв полотна;
- після клацання – зникати.

Крім того, у класі мають підраховуватися кількості екземплярів-бульбашок, наявних і зловлених.

На кроках 3-5 наведено, як створити клас **Bubble**, екземпляри якого поводитимуться на полотні **canvas** вікна **tk** саме так, як описано.

3. Конструктор **__init__**:

```
class Bub():
```

```

number = 0 # Лічильник бульбашок на полотні
clicked = 0 # Лічильник клацнутих бульбашок
def __init__(B):

    B.d = 40
    B.x = randint(2,755)
    B.y = randint(2,455)
    B.dx = randint(-5, 5)
    B.dy = randint(-5, 5)
    Bub.number = Bub.number+1

    B.id = canvas.create_oval(B.x, B.y, B.x+B.d, B.y+B.d, fill =
        'blue')
    canvas.tag_bind(B.id, '<1>', B.on_click) # Бульбашка має
        обробник
    B.move() # клацання і починає рухатись після
        створення

```

4. Метод **move**, що забезпечує рух і відбивання кульки:

```

def move(B):
    canvas.move(B.id, B.dx, B.dy)
    tk.update()
    pos = canvas.coords(B.id)
    if len(pos)>0: # Якщо кулька ще не зникла...
        if pos[0]<2 or pos[2]>798: # ...то відбиваємо від країв
            B.dx = -1*B.dx
        if pos[1]<2 or pos[3]>498:
            B.dy = -1*B.dy
    tk.after (100, B.move) # Продовження руху

```

5. Метод-обробник клацання бульбашки **on_click**:

```

def on_click(B,event):
    if Bub.number<maxB:
        canvas.delete(B.id)
        Bub.number= Bub.number-1
        Bub.clicked = Bub.clicked+1

```

```
tk.title(Bub.clicked)
```

6. Створіть функцію **play()**, яка забезпечить створення бульбашок у потрібні моменти, а також вчасне припинення гри:

```
def play():
```

```
    if Bub.number<maxB:      # Якщо бульбашок мало...
        Bub()                # -то створюємо бульбашку
        tk.after(genT, play)  # продовжуємо створювати
```

```
    else:
```

```
        tk.after_cancel(play) # Припиняємо створювати
        tk.title('Гру закінчено. Ваш результат:' + str(Bub.clicked))
```

7. В основній програмі запишіть команди виклику функції

```
play():
```

```
play()
```

```
tk.mainloop()
```

Збережіть програмний код у файлі з іменем **Завдання 22** і випробуйте програму. За потреби виправте помилки. Коли програма повністю запрацює, додайте до неї можливості, описані в наступних пунктах.

8. Зараз діаметр всіх бульбашок однаковий. Зробіть так, щоб бульбашка під час створення отримувала випадкове значення діаметра в межах від 20 до 40 пікселів.
9. По завершенні гри бульбашки, що залишились, продовжують рухатись, але не реагують на клацання. Зробіть так, щоб клацання в цей період спричиняло змінення кольору на жовтий ('Yellow').
10. Кожна бульбашка під час клацання до загального рахунку додає 1 бал. Зробіть так, щоб швидкі бульбашки малого розміру додавали по 2 бали.
11. Бульбашки з'являються через рівні проміжки часу, визначені змінною **genT** (початкове значення 1000 мс). Зробіть так, щоб цей інтервал поступово зменшувався.
12. Запропонуйте власну ідею покращення програми і реалізуйте її.

Урок 23. Поняття, реалізація та застосування масивів (списків).

Завдання. Скласти ігрову програму *Відгадай столицю*. Занести назви країн і їхніх столиць до двох списків. Програма має по чергово запитувати назву столиці і перевіряти правильність відповіді, а після трьох помилок припиняти роботу, повідомляючи результат.

1. У **Python IDLE** виберіть команду **File** \Rightarrow **New File**.

2. Створіть список назв країн і список їхніх столиць:

```
kr = ['Франція', 'ОАЕ', 'Нідерланди', 'Туреччина',  
      'Швейцарія']
```

```
st = ['Париж', 'Абу-Дабі', 'Амстердам', 'Анкара', 'Берн']
```

Створіть змінну-лічильник помилок: $k = 0$. Надайте значення 0 змінній i для збереження індексу поточного елемента.

```
k = 0
```

```
i = 0
```

3. Запитання пропонуються доти, поки не закінчився список країн ($i < 5$) і користувач не припустився третьої помилки ($k < 3$).

```
while k < 3 and i < len(kr):
```

```
    print('Країна: ', kr[i])
```

```
    s = input('Назвіть столицю країни: ')
```

```
    if s == st[i]:
```

```
        print('Правильно!')
```

```
    else:
```

```
        k = k + 1
```

```
        print('Помилка!')
```

```
    i = i + 1
```

4. Напишіть оператор розгалуження, в якому залежно від значення k виводиться відповідне повідомлення:

```
if (k == 0):
```

```
    print('Вітаю! Ви – знавець географії!')
```



```

elif(k == 1):
    print('Непогано! Ви можете вирушати в подорож!')
elif(k == 2):
    print('Почитайте підручник географії!')
else:
    print('Вам краще сидіти дома!')

```

5. Додайте до списків назви ще декількох країн і столиць. Внесіть до програмного коду відповідні зміни.
6. Збережіть програму у файлі з іменем **Завдання 23**.
7. Випробуйте програму, проаналізуйте результати її роботи.

Урок 24. Функції та методи опрацювання масивів (списків)

Завдання 1. Розробити програму опрацювання списку значень середньодобової температури за один тиждень місяця. Визначити, скільки разів за тиждень значення температури змінювало знак.

1. У **Python IDLE** виберіть команду **File** ⇒ **New File**.
2. Завантажте модуль для генерації випадкових чисел:

```
from random import*
```
3. Створіть порожній список *t*.
4. Заповніть список випадковими значеннями в діапазоні від -10 до +5. Виведіть утворений список у консоль.

```
for i in range(10):
    a=randint(-10,5)
    t.append(a)
print(t)
```
5. Створіть змінну *k* для збереження кількості змін знака температури.
6. Запишіть оператор циклу для перегляду елементів списку з індексами від 0 до 5.

Якщо добуток поточного і наступного елементів від'ємний, потрібно вивести індекси цих елементів і збільшити значення лічильника k :

```
k = 0
for i in range(6):
    if(t[i]*t[i+1]<0):
        print(i, ' ', i+1)
        k += 1
```

7. Запишіть оператор для виведення значення k . Збережіть програму у файлі з іменем **Завдання 24**.
8. Випробуйте програму. Внесіть зміни до програми, щоб окремо підраховувалось число днів, коли середньодобова температура була рівна 0.

Завдання 2. Скласти програму опрацювання списку. Дано список оцінок, отриманих учнями за проходження тесту.

Складіть програму для визначення:

- а) середнього арифметичного оцінок;
- б) кількості учнів, які отримали оцінку, вищу за середній бал.

1. У **Python IDLE** виберіть команду **File** \Rightarrow **New File**.
2. Створіть список оцінок:
`a = [11, 8, 6, 8, 12, 9, 7, 6]`
3. Обчисліть s – суму елементів списку a .
4. Обчисліть середнє арифметичне елементів списку a , збережіть це значення в змінній av .
5. Додайте оператор виведення результату, округленого до цілого за допомогою функції **round()**:
`print (round(av))`
6. Запишіть програмний код для визначення кількості елементів списку, більших за значення середнього арифметичного.
7. Збережіть програму у файлі з іменем **Завдання 24_2**. Випробуйте програму, проаналізуйте результати її

роботи. Доповніть програму командами для введення списку з клавіатури й випробуйте її.

Урок 25. Текстові рядки як масиви символів

Завдання. У XIX ст. для пересилання важливих повідомлень використовувались шифри, в яких літери переставляються за певним правилом. Напишіть програму, що шифрує повідомлення, міняючи місцями кожні два знаки.

Наприклад: введення: Я чекатиму на тебе завтра

виведення: Ячечакитумн аеттебз варт а

1. У **Python IDLE** виберіть команду **File** ⇒ **New File**.
2. Запрограмуйте введення значення змінної *s*. Передбачте виведення підказки "Введіть текст для шифрування".
3. Створіть змінну *s1* для формування результату і запишіть у неї порожній рядок.
4. Число символів у рядку *s* має бути парним, тому запишіть оператор розгалуження для реалізації такої дії: якщо довжина рядка *s* є непарним числом, то додати до рядка пропуск:

```
if(len(s)%2 != 0):  
    s = s+' '
```

5. Запишіть оператор циклу для перебору непарних індексів.

В тілі циклу до рядка *s1* додаються символи *s[i+1]* і *s[i]*:

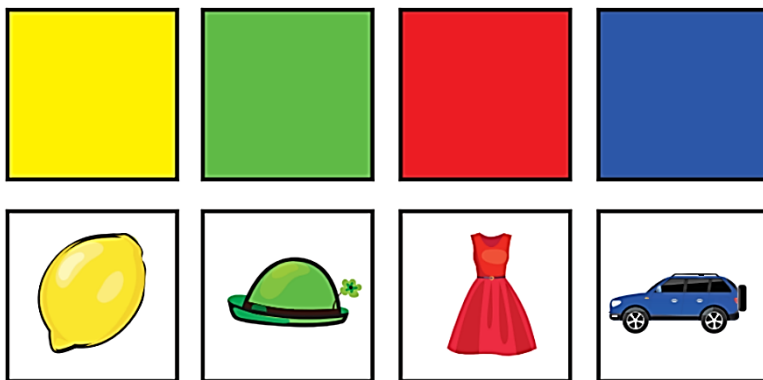
```
for i in range(0, len(s), 2):  
    s1 = s1+s[i+1]+s[i]
```

6. Виведіть зашифроване повідомлення.
7. бережіть програмний код у файлі з іменем **Завдання 25**.
8. Випробуйте програму. Переконайтесь, що повідомлення успішно розшифровується за допомогою цієї ж програми.

Урок 26. Зображення як цілісний об'єкт та як масив пікселів

Вправа. Об'єкти та їхні кольори.

Обговоріть, які засоби мови програмування Python можна використати для реалізації проєкту **Об'єкти та їхні кольори**. На екрані користувачам пропонуються чотири квадрати деякого кольору, під ними – об'єкти, основний колір яких відповідає кольору геометричної фігури. При натисканні на квадрат під зображенням з'являється надпис англійською мовою з назвою об'єкта.



Урок 27. Відео як цілісний об'єкт та як масив зображень. Анімації

Завдання. Створимо анімацію Політ метелика із трьох кадрів. Розглянемо кроки створення анімації.

1. Підготуйте три графічних файли з зображеннями фаз руху метелика. Імена цих файлів зберігатимуться у списку **anim**.



2. Введіть програмний код:

```
import time  
from tkinter import *
```

```

def play():
    for i in range(3):
        canvas.create_image(0, 0, anchor = NW, image =
            frames[i])
        root.update()      # Оновлення полотна
        time.sleep(0.1)    # Пауза на 0.1 секунди
        root.after(0, play) # Повторний виклик функції play()
root = Tk()
root.title('Політ метелика')
root.geometry('200x250')
canvas = Canvas(root, width = 200, height = 200)
canvas.place(x = 1, y = 1)
anim=['1.gif', '2.gif', '3.gif'] # Список імен файлів
frames = []                      # Читаємо дані зображень із файлів,..
for name in anim:                # імена яких у списку anim,..
    frames.append(Photolmage(file = name)) # у список frames
play()
root.mainloop()

```

Урок 28. Бібліотеки для роботи з мультимедійними даними

Завдання. Скласти програму **Калькулятор** для обчислення результату виконання арифметичних дій над двома введеними числами.

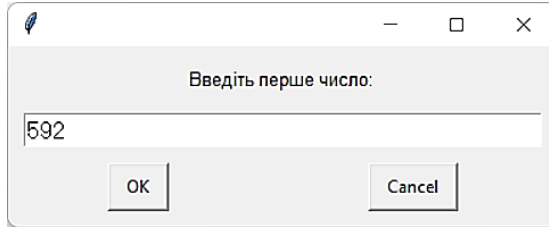
1. У **Python IDLE** виберіть команду **File** ⇒ **New File**.
2. Запишіть оператор для завантаження модуля **easygui**:

```
from easygui import*
```
3. Уведення чисел і обчислення виразу відбуватимуться в циклі **while(rep)**, тобто поки значення змінної **rep** дорівнює **True**. Задайте початкове значення змінної **rep**:

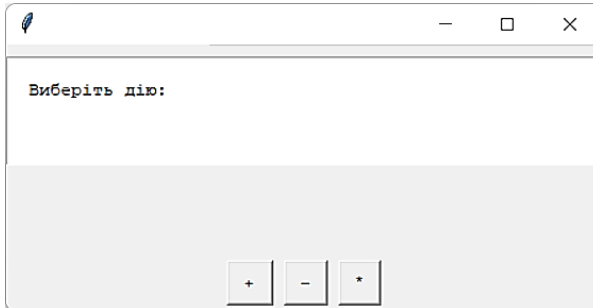
```
rep = True
```
4. У тілі циклу **while(rep)**: введіть два числа за допомогою діалогового вікна **enterbox()**:

```
while(rep):
    n1 = float(enterbox('Введіть перше число: '))
```

```
n2 = float(enterbox('Введіть друге число: '))
```



5. Реалізуйте вибір арифметичної дії за допомогою відповідних кнопок діалогового вікна **buttonbox()**



Вибраний елемент списку присвойте змінній operation:

```
ch = ['+', '-', '*']
```

```
operation = buttonbox('Виберіть дію: ', choices = ch)
```

Проаналізуйте значення змінної operation і організуйте виконання потрібної арифметичної дії в операторі if ... elif ... else і виведення результату:

```
if operation == '+':
```

```
    msgbox('Результат ' + str(n1+n2))
```

```
elif operation == '-':
```

```
    msgbox('Результат ' + str(n1-n2))
```

```
else:
```

```
    msgbox('Результат ' + str(n1*n2))
```

6. Щоб запитати в користувача, чи слід повторити обчислення, створіть діалогове вікно з кнопками Так, Ні; вибраний елемент списку присвойте змінній **povtor**:
- ```
povtor = buttonbox('Ще раз? ', choices = ['Так', 'Ні'])
```

7. Якщо змінна `povtor` набуває значення `Hi`, змінна `rep` отримує значення `False`, і цикл `while(rep)`: припиняє роботу:

```
if povtor == 'Hi':
 rep = False
```

8. Додайте до списку заголовків кнопок діалогового вікна для вибору дії кнопки «/», «//», «%» і запрограмуйте відповідні обчислення. Запропонуйте власне розширення можливостей калькулятора і реалізуйте його. Збережіть програму в файлі з іменем **Завдання 28**.

### **Урок 29. Графічний інтерфейс користувача.**

Вправа 1. Повідомлення про необхідність дотримання часу роботи за комп'ютером.

*Завдання.* Ваш молодший брат любить довго засиджуватися за комп'ютером. Створіть програму, після запуску якої буде з'являтися вікно з повідомленням про необхідність дотримання часу роботи за комп'ютером. Доберіть властивості напису, які привернуть увагу брата.

Вправа 2. Планування діяльності своєї родини.

*Завдання.* Ви хочете застосувати свій комп'ютер для планування діяльності своєї родини. Сплануйте та створіть прикладну програму, після запуску якої у вікні з'являтиметься повідомлення, актуальне для вашої родини, наприклад, нагадування про день народження, важливу подію, завдання, яке слід виконати тощо.

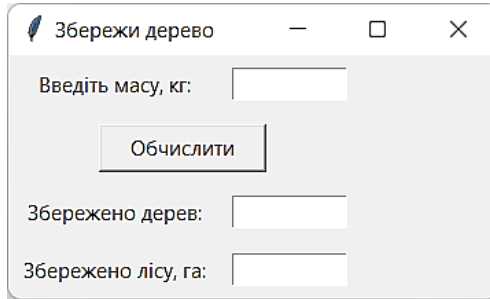
Вправа 3. Календар.

*Завдання.* Створіть програму, після запуску якої на екранній формі буде відображено календар на поточний місяць. Скористайтесь підказкою: в середовищі програмування в контекстному меню команд можна використовувати команди для роботи з фрагментами коду.

### **Урок 30. Прикладний програмний інтерфейс.**

*Завдання.* Написати програму для розв'язування задачі:

*Задача.* Створити програму **Збережи дерево**, яка за масою макулатури визначає, скільки дерев і яку площу лісу допоможе зберегти від знищення її переробка. Відомо, що переробка 60 кг макулатури зберігає 1 дерево і 0,0005 га лісу.



1. У **Python IDLE** виберіть команду **File** ⇒ **New File** з назвою **Завдання 30**. Створіть вікно програми із заголовком **Збережи дерево**. Завершіть програму оператором `root.mainloop()`.
2. Створіть віджети класу **Label** і **entry1** класу **Entry** для введення даних:
 

```
Label(text = 'Введіть масу, кг:').grid(row = 0, column = 0,
 padx = 8, pady = 8)
entry1 = Entry(width = 10)
entry1.grid(row = 0, column = 1, padx = 8, pady = 8)
```
3. Розмістіть у вікні віджет **b1** класу **Button**:
 

```
b1 = Button(text = 'Обчислити', command = b1_click, width = 14)\
.grid(row = 1, column = 0, columnspan = 2, padx = 8, pady = 8)
```
4. Додайте решту написів та поля **entry1**, **entry2** згідно з зразком.
5. Опишіть функцію **b1\_click()** для обчислення кількості збережених дерев і площі врятованого лісу:
 

```
def b1_click():
 n = int(entry1.get())
 entry2.delete(0, END)
 entry2.insert(0, str(int(n/60)))
```



```
entry3.delete(0, END)
entry3.insert(0, str(n/60*0.0005))
```

6. Запустіть програму для різних початкових значень. Проаналізуйте її роботу.

### **Урок 31. Дані сенсорів та датчиків.**

**Завдання.** У класі створюється рейтинг дітей за підсумками семестру. Якщо середній бал семестрових оцінок учня/учениці більше за 10, то він/вона отримує 10 рейтингових балів, якщо більше за 7 і не більше за 10, – 7 балів, якщо більше за 4 і не більше за 7, – 4 бали, якщо не більше за 4, – 1 бал. Крім того, учень/учениця отримує по 5 додаткових балів, якщо він/вона відвідує шкільні предметні гуртки, шкільні спортивні секції, шкільну театральну студію. Створіть проєкт для визначення рейтингового балу учня/учениці.

1. У **Python IDLE** виберіть команду **File** ⇒ **New File** з назвою **Завдання 31**.
2. Уведіть команди створення вікна розміром 400 × 450 пікселів.
3. Уведіть команди створення та розміщення у вікні проєкту чотирьох перемикачів, які визначатимуть, який середній бал має учень/учениці. Для цього:

- 1) Створіть змінну, яка буде пов'язана з усіма перемикачами:

```
choice = IntVar()
```

- 2) Уведіть команди створення та розміщення перемикача з іменем `switch1`. Значення змінної `choice` для нього 1:

```
switch1 = Radiobutton(text = 'Більше за 10', variable =
 choice, value = 1)
```

```
switch1.pack(pady = 10)
```

- 3) Уведіть команди створення та розміщення ще трьох перемикачів з іменами `switch2`, `switch3`, `switch4`.

Значення змінної choice для них 2, 3 та 4 відповідно, значення властивості text, як наведено на малюнку.

4. Уведіть команди створення та розміщення у вікні трьох прапорців, які визначатимуть, чи відвідує учень/учениця шкільні предметні гуртки, шкільні спортивні секції, шкільну театральну студію. Для цього:

- 1) Створіть змінну choice1, яка буде пов'язана з першим прапорцем:

```
choice1 = BooleanVar()
```

- 2) Уведіть команди створення та розміщення першого прапорця з іменем flag1, пов'язаного зі змінною choice1:

```
flag1 = Checkbutton(text = 'Предметні гуртки', variable =
choice1)
```

```
flag1.pack(pady = 10)
```

- 3) Уведіть команди створення та розміщення ще двох прапорців з іменами flag2 та flag3. Імена змінних для них choice2 та choice3 відповідно, значення властивості text, як наведено на малюнку.

5. Уведіть команди створення та розміщення у вікні напису для виведення сумарного рейтингу, напису для пояснювального тексту з іменем lab2 і кнопки.

Установіть значення їх властивостей, як це наведено на малюнку.

6. Запустіть проєкт на виконання, перевірте, як працюють компоненти перемикачі та прапорці.
7. Закрийте вікно виконання проєкту.
8. Створіть процедуру – обробник події Click для кнопки та додайте до неї такі команди:

```
s = 0
```

```
if choice.get() == 1: s = 10
```

```
elif choice.get() == 2: s = 7
```

```
elif choice.get() == 3: s = 4
```

```
elif choice.get() == 4: s = 1
```

```
if choice1.get(): s += 5
```

```
if choice2.get(): s += 5
if choice3.get(): s += 5
lab2['text'] = str(s)
```

9. Збережіть оновлену версію проєкту.
10. Запустіть проєкт на виконання.
11. Виберіть перший перемикач, перший і третій прапорці.
12. Виберіть кнопку.
13. Проаналізуйте правильність отриманого результату.
14. Виберіть інший перемикач та інші прапорці.
15. Проаналізуйте правильність отриманого результату.

### **Урок 32. Зовнішні джерела даних.**

**Завдання.** Написати програму, що підраховує кількість заданих символів у фрагменті тексту.

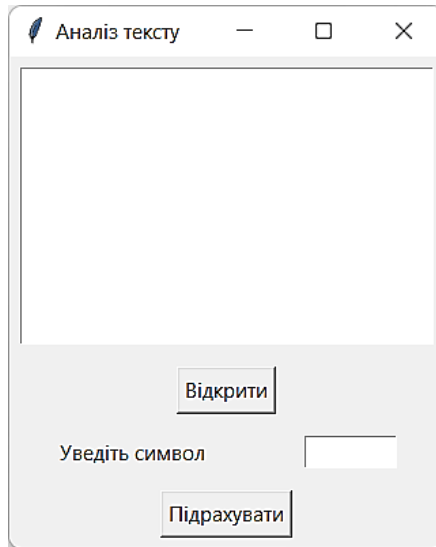
1. Створіть текстовий файл **f\_text.txt** і запишіть у нього фрагмент статті з вікіпедії про вашу область.
2. У **Python IDLE** виберіть команду **File** ⇒ **New File**, збережіть файл з назвою **Завдання 32**.
3. Завантажте модулі **tkinter**, **filedialog** (як **fd**), **messagebox**.

```
from tkinter import*
from tkinter import filedialog as fd
from tkinter import messagebox
```

4. Створіть вікно із заголовком **Аналіз тексту**. Розмістіть у вікні віджет **entry** класу **Entry** для задання шуканого символу, віджет **text** класу **Text** – для показу вмісту файлу **f\_text.txt**.

```
root=Tk()
root.title('Аналіз тексту')
text=Text(width='30', heigh='10')
text.grid(row = 0, column = 0, columnspan = 2, padx = 8, pady = 8)
```

5. Додайте до вікна два віджети **Button**: для відкриття файлу і для підрахунку числа входжень шуканого символу.



6. Опишіть функцію **loadText()** – обробник події для кнопки **b\_open**:

```
def loadText():
 file_name = fd.askopenfilename()
 f = open(file_name)
 text.insert(1.0, f.read())
 f.close()
```

7. Опишіть функцію **count\_C()** – обробник події для кнопки **b\_count**:

```
def count_C():
 c = entry.get()
 k = text.get(1.0, END).count(c)
 messagebox.showinfo('Відповідь', 'k = '+str(k))
```

8. Випробуйте роботу програми для різних значень символу **c**.

### ***Урок 33. Поняття та приклади інтерактивних інсталяцій.***

**Завдання.** Створити довідник «Міста України». Залежно від вибору перемикача виводити до текстового поля інформацію про місто.

1. Підготуйте декілька текстових файлів з інформацією про міста України. Збережіть файли в папку **Міста України**.
2. У **Python IDLE** виберіть команду **File** ⇒ **New File**, збережіть файл з назвою **Завдання 33** в папку **Міста України**.
3. Створіть вікно програми із заголовком **Міста України**. Додайте віджет **text1** класу **Text**:

```
text1 = Text(width = 24, height = 10, wrap = WORD)
```

4. Створіть перемикач із 5 радіокнопок із підписами, що відповідають назвам міст:

```
var = IntVar()
```

```
var.set(0)
```

```
kyiv = Radiobutton(text = 'Київ', variable = var, value = 0,
 command = change)
```

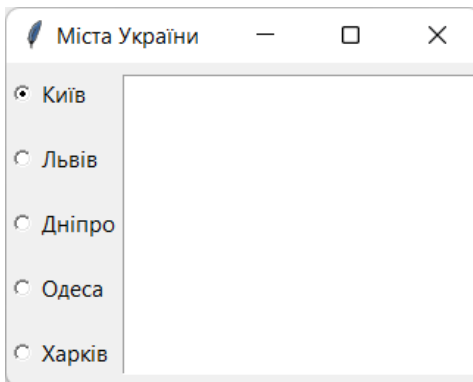
```
lviv = Radiobutton(text = 'Львів', variable = var, value = 1,
 command = change)
```

```
<...>
```

```
kyiv.grid(row = 0, column = 0, sticky = W)
```

```
lviv.grid(row = 1, column = 0, sticky = W)
```

```
<...>
```



5. Опишіть спільний обробник події для перемикачів – функцію **change()**. Залежно від значення **var.get()** зі списку вибирається ім'я файлу з інформацією про вибране місто:

```
def change():
 files = ['Київ.txt', 'Львів.txt', 'Дніпро.txt', 'Одеса.txt',
 'Харків.txt']
 view(files[var.get()])
```

6. Опишіть функцію **view()**, у якій до поля віджета **text1** виводиться вміст файлу з назвою з аргументе **filename**.

```
def view(file_name):
 text1.delete(1.0, END)
 f = open(file_name)
 text1.insert(1.0, f.read())
 f.close()
```

7. Додайте кілька перемикачів із назвами ваших улюблених міст, підготуйте повідомлення про них. Випробуйте програму.

### ***Урок 34. Виконання навчальних проєктів.***

Виконати навчальний проєкт «Калькулятор».

### ***Урок 35. Представлення та захист навчальних проєктів.***

Захист навчальних проєктів можна провести у формі предметних турнірів з доповідачем, опонентами та рецензентами.

### **Завдання для тестування<sup>3</sup>.**

#### ***Тест до теми: Цифрове мистецтво та творчість***

##### **1. Цифрове мистецтво - це ...**

- галузь людської культури, в якій за допомогою знаків через конкретні образи світу виражаються його узагальнені сенси
- мистецтво відображення суцього у вигляді різних образів, зокрема таких як художні образи на площині (графіка, малярство тощо) та в просторі (скульптура).

---

<sup>3</sup> Правильні відповіді відмічено жирним шрифтом.

- **напря́м в ме́діамистецтві, твори якого створюються і представляються за допомогою сучасних інформаційно-комунікаційних або ме́діа технологій, результатом якого є художні твори в цифровій формі.**
2. До інтерактивних творів належать:
- сукупність писаних і друкованих творів певного народу, епохи, людства; різновид мистецтва слова, який описує дійсність художніми образами
  - твори живопису, скульптури, графіки, декоративно-прикладного і народного мистецтва, що зображують тварин.
  - **веб-ресурси, мультимедійні підручники, мультимедійні презентації, комп'ютерні ігри, інтерактивні галереї, музичні твори**
3. Дизайн - це ...
- **це творча діяльність із проектування естетичних властивостей предметів і середовища, в якому живе людина.**
  - це фахівець, що відповідає за функціональний та естетичний рівень предметів та композицій із них, створюючи певне середовище.
  - мистецтво зображення певного предмета, візерунка, портрета або явища на площині різноманітними художніми техніками.
4. Промисловий дизайн - це ...
- галузь дизайну, спрямована на інтер'єр приміщень з метою забезпечити зручність і естетично приємну взаємодію середовища з людьми.
  - **сфера проектної діяльності, яка займається художнім проектуванням елементів предметного наповнення середовища проживання людини, створені засобами індустріального виробництва.**

- проектування освітлення для забезпечення комфорту, естетичного зорового сприйняття, енергоефективності.
- акваріум, тераріум, флораріум; фонтани, водоспади, ставочки, композиції з натурального і штучного каменю, зимові сади й настінні садочки, куточки з рослинами, флористичні композиції.

#### 5. Біодизайн - це ...

- галузь дизайну, спрямована на інтер'єр приміщень з метою забезпечити зручність і естетично приємну взаємодію середовища з людьми.
- сфера проектної діяльності, яка займається художнім проектуванням елементів предметного наповнення середовища проживання людини, створені засобами індустріального виробництва.
- проектування освітлення для забезпечення комфорту, естетичного зорового сприйняття, енергоефективності.
- **галузь дизайну, спрямована на створення акваріумів, тераріумів, флораріумів; фонтанів, водоспадів, ставочків, композицій з натурального і штучного каменю, зимових садів й настінних садочків, куточків з рослинами, флористичних композицій.**

#### 6. Дизайн інтер'єру - це ...

- **галузь дизайну, спрямована на інтер'єр приміщень з метою забезпечити зручність і естетично приємну взаємодію середовища з людьми.**
- сфера проектної діяльності, яка займається художнім проектуванням елементів предметного наповнення середовища проживання людини, створені засобами індустріального виробництва.
- проектування освітлення для забезпечення комфорту, естетичного зорового сприйняття, енергоефективності.



- акваріум, тераріум, флораріум; фонтани, водоспади, ставочки, композиції з натурального і штучного каменю, зимові сади й настінні садочки, куточки з рослинами, флористичні композиції.

#### 7. Що таке Processing?

- мова програмування високого рівня, метою створення якої було отримати просту в користуванні мову для початківців.
- алгоритмічна мова програмування універсального призначення.
- **мова програмування, створена в 2002 році для роботи графічними об'єктами - геометричними фігурами та зображеннями.**
- інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією.

#### 8. Програмування - це ...

- **це є як написання коду, так і спосіб думати, як зробити щось нове та унікальне, а також обійти деякі обмеження існуючих програмних засобів.**
- розділ інформатики, метод опису систем або процесів шляхом створення алгоритмів їх функціонування.
- сукупність організаційних, правових, політичних, соціально-економічних, науково-технічних, виробничих процесів, що спрямовані на створення умов для задоволення інформаційних потреб громадян та суспільства на основі створення, розвитку і використання інформаційних систем, мереж, ресурсів та інформаційних технологій, які побудовані на основі застосування сучасної комп'ютерної та мережевої техніки.

#### 9. Скетчінг - це ...

- процес перетворення фактів, інформації, емоцій тощо на цілісне й упорядковане знання.

- **це спосіб мислити. Він ефективний і схожий на гру. Основна мета - оцінити велику кількість ідей за короткий проміжок часу.**
- психологічний стан, який виражає собою правильність ухваленого рішення і супроводжуваний відчуттям упевненості в точності сприйняття або інтерпретації якої-небудь події, явища, факту.

10. Класифікація мов програмування за принципом програмування:

- **Процедурні • Непроцедурні • Об'єктноорієнтовані**
- Універсальні • Спеціалізовані
- Низького рівня • Високого рівня

11. Модуль - це ...

- послідовність правил і операцій для вирішення певної задачі.
- послідовність інструкцій, написаних мовою програмування, яка реалізує алгоритм вирішення задачі.
- **готовий фрагмент програмного коду, який може бути приєднаний до нових проєктів, що спрощує та пришвидшує їх створення**

12. Для чого призначено програму-компілятор (програму-інтерпретатор)?

- перетворює (компілює) вихідний код, написаний певною мовою програмування, на семантично еквівалентний код в іншій мові програмування, який зазвичай необхідний для виконання програми машиною, наприклад, комп'ютером.
- Для перегляду файлів, тек, периферійних пристроїв.
- **Ця програма по чергово аналізує команди проєкту на наявність синтаксичних помилок.**

## *Тест до теми «Мова програмування. Середовище програмування»*

1. Система позначень для точного опису алгоритму, який потрібно виконати за допомогою комп'ютера це...

- **Програма**
- Алгоритм
- Мова програмування
- Виконавець

2. Графічний спосіб подання алгоритмів передбачає використання спеціальних графічних позначок, це...

- **блок-схеми**
- довільні зображення
- програма
- ярлик програми

3. Система програмування складається з таких компонентів (6 шт):

- **транслятор**
- **текстовий редактор**
- **компонувальник**
- **бібліотека стандартних програм**
- **налагоджувач**
- **візуальні засоби автоматизації програмування**
- система голосових підказок
- система автоматичного створення програм відповідно до наданої документації

4. Зчитує одразу всю програму й переписує її машинним кодом або мовою асемблер, результат перекладу зберігається в окремому файлі.

- Константа
- Змінна
- Функція
- **Компілятор**
- Інтерпретатор

5. Зчитує вихідний код програми по одній команді й одразу намагається їх «перекладати» та виконувати. Це дає змогу програмісту швидше перевіряти правильність виконання програми та знаходити помилки в коді
- Константа
  - Змінна
  - Функція
  - Компілятор
  - **Інтерпретатор**
6. Мова програмування має такі складові:
- **Словник**
  - **Синтаксис**
  - **Алфавіт**
  - **Семантика**
  - Каталог файлів
  - Засіб встановлення операційної системи
7. Набір символів, з яких утворюють слова та речення цієї мови
- Словник
  - Синтаксис
  - **Алфавіт**
  - Семантика
8. Сукупність спеціальних слів, які мають однозначне пояснення та застосування
- **Словник**
  - Синтаксис
  - Алфавіт
  - Семантика
9. Система правил складання базових конструкцій мови
- Словник
  - **Синтаксис**
  - Алфавіт
  - Семантика

10. Правила, що пояснюють, яке смислове значення має опис кожної з вказівок програми і які дії повинен виконати комп'ютер під час виконання кожної з таких вказівок
- Словник
  - Синтаксис
  - Алфавіт
  - Семантика
11. За технологією програмування розрізняють на...
- процедурні
  - об'єктно-орієнтовні
  - декларативні
  - мова скриптів (мова сценаріїв)
  - інтерпретаторні
  - компіляторні
  - трансляторні
12. Оберіть правильний перелік об'єктно-орієнтованих мов програмування
- **Python, Java, C#, C++, Delphi**
  - JavaScript, Python, Java, C#, C++
  - JavaScript, Python, Java, C#, PHP
  - JavaScript, Python, Java, C#, Prolog.

### *Тест до теми «Мова Python. Цикл for»*

1. Для того щоб описати повторення визначену кількість разів, використовують оператор циклу...
- з параметром
  - з передумовою
  - з післяумовою
  - будь-який з перелічених вище
2. Мовою Python цикл з лічильником правильно записується таким чином:

```
for i := 1 to 5 do
begin
 a := 3*a + 1;
 p := p + a;
end;
```

- Так
- Ні

3. for i in \_\_\_\_\_:

Що повинно бути вказано в рядку, який розпочинається службовим словом for замість підкреслення?

- шкала
- команда циклу
- структура розгалуження
- ім'я файлу

4. Для того, щоб була побудована шкала: 0, 1, 2, 3, 4, 5, можна використати такі команди:

- **range(6)**
- range(5)
- **range(0, 6)**
- range(0, 5)
- **range(0, 6, 1)**
- range(0, 1, 6)

5. Шкала, побудована функцією range(7,18,2), матиме найбільше значення:

- 15
- 16
- **17**
- 18
- 23

6. Який підрахунок виконає така програма?

```

k=int(input("Введіть перше число="))
n=int(input("Введіть друге число="))
s=0
for i in range(k,n+1):
 if (i%2==0):
 s=s+i
print(s)

```

- Підрахунок суми чисел від k до n
- **Підрахунок суми парних чисел від k до n**
- Підрахунок суми непарних чисел від k до n
- Друкування чисел в стовпчик з діапазону від k до n
- Друкування чисел в рядок з діапазону від k до n

7. Така програма виконає виведення на екран чисел...

```

for chislo in range(5,-10,-3) :
 print(chislo)

```

- **від 5 до -10 з кроком -3 (у стовпчик)**
- від 5 до -10 з кроком -3 (у рядок)
- від -10 до 5 з кроком -3 (у стовпчик)
- від -10 до 5 з кроком 5 (у стовпчик)

8. Скільки рядків видрукує дана програма?

```

for i in range(1,11):
 print('2 *', i, '=', 2*i)

```

- 1
- 2
- **10**
- 11
- 20

9. Скільки рядків видрукує дана програма?

```

for n in range(1,11):
 for i in range(1,11):
 k=n*i
 print(n, '*', i, '=', k)

```

- 1
- 11
- 22
- **100**
- 111
- 121

10. Що буде виводити на екран дана програма?

```
n=int(input("Введіть число "))
for i in range(1,n+1):
 if (n%i==0):
 print (i)
```

- Числа, кратні числу n
- **Дільники числа n**
- Відсоток, що складає число n від числа i
- Ранг матриці розмірності n x i

11. Дана програма виведе результат...

```
for i in 'hello world':
 if i == 'a':
 break
 else:
 print ('такої літери немає')
```

- hello world
- break
- else:
- print ('такої літери немає')
- **такої літери немає**

12. Оберіть неправильний варіант:

- **значення n функції range (n) може бути дробовим числом**
- значення n функції range (n) має бути цілим числом
- шкала, побудована функцією range (n) починається з 0
- шкала, побудована функцією range (n) має крок 1



## *Тест до теми «Графічний інтерфейс Python», ч. 1*

1. Для підключення модуля tkinter необхідно написати в середовищі програмування такий код:

- **from tkinter import\***
- from tkinter import
- tkinter import
- tkinter import\*

2. Оберіть властивості головного вікна програми (4 шт.)

- **title**
- **geometry**
- **minsize**
- **maxsize**
- pack
- grid
- label
- button

3. Властивість root.title () встановлює заголовок головного вікна програми

- **Так**
- Ні

4. Встановіть послідовність операторів, які потрібні для створення вікна програми.

1. root.geometry('700x400')
2. root.mainloop()
3. root=Tk()
4. from tkinter import\*

**Відповідь: 4 3 1 2**

5. Встановіть відповідність між віджетами та класом до якого вони належать:

Label Кнопка  
Button Напис

6. Метод `mainloop()` забезпечує постійну роботу головного вікна і його об'єктів до моменту, коли воно буде.....

- **відкрито**
- закрито

7. Команда `btn.bind('<Button-1>', rez)`

- **Викликає функцію `rez` після натиснення на кнопці ЛІВОЮ клавішою миші**
- Викликає функцію `rez` після натиснення на кнопці ПРАВОЮ клавішою миші
- Викликає функцію `rez` після натиснення на КОЛЕСО ПРОКРУТКИ миші

8. Команда `btn.bind('<Button-3>', rez)`

- Викликає функцію `rez` після натиснення на кнопці ЛІВОЮ клавішою миші
- **Викликає функцію `rez` після натиснення на кнопці ПРАВОЮ клавішою миші**
- Викликає функцію `rez` після натиснення на КОЛЕСО ПРОКРУТКИ миші

### *Тест до теми «Графічний інтерфейс Python», ч. 2*

1. Відображає текст у вікні і служить в основному для інформаційних цілей (вивід повідомлень, підпис інших елементів інтерфейсу тощо)...

- **елемент керування "Label"**
- елемент керування "Button"
- елемент керування "Entry"
- елемент керування "Text"
- елемент керування Scrollbar

2. Служить в основному для введення рядка тексту....

- елемент керування "Label"
- елемент керування "Button"
- **елемент керування "Entry"**
- елемент керування "Text"

- елемент керування Scrollbar
3. Самий непередбачуваний з наявних пакувальників. Як правило цей пакувальник використовують для розташування віджетів один за одним (зліва направо або зверху донизу)
    - **pack**
    - grid
    - place
    - widget
  4. Розташовує віджет у комірку прямокутної таблиці...
    - pack
    - **grid**
    - place
    - widget
  5. Розміщує віджет у фіксованому місці з фіксованим розміром. Також дозволяє вказувати координати розташування у відносних одиницях для реалізації «гумового» розміщення.
    - pack
    - grid
    - **place**
    - widget
  6. Якою командою віджету вказується його положення в частках батьківського вікна, тобто відносно?
    - root.geometry('200x150')
    - lab1 = Label(bg='blue', width=7, height=3)
    - lab1.place(x=75, y=10)
    - lab2 = Label(bg='yellow', width=7, height=3)
    - **lab2.place(relx=0.3, rely=0.5)**
  7. Якою командою віджету вказується його положення в абсолютних значеннях (в пікселях)?
    - root.geometry('200x150')

- lab1 = Label(bg='blue', width=7, height=3)
- **lab1.place(x=75, y=10)**
- lab2 = Label(bg='yellow', width=7, height=3)
- lab2.place(relx=0.3, rely=0.5)

8. Які команди надають змінним числових значень?

- **a=float(ent1.get())**
- d=ent2.get()
- **b=float(ent3.get())**
- lab4.grid(row=1, column=2)
- btn1.grid(row=2, column=2)

9. У tkinter за допомогою методу bind() між собою зв'язуються віджет, подія і дія. Можливі такі події:

|                                                        | <Button-1><br>або <1> | <Button-2><br>або <2> | <Button-3><br>або <3> |
|--------------------------------------------------------|-----------------------|-----------------------|-----------------------|
| натиснута ліва<br>клавіша миші                         | +                     |                       |                       |
| натиснута права<br>клавіша миші                        |                       |                       | +                     |
| натиснута середня<br>клавіша миші (або<br>колесо миші) |                       | +                     |                       |

10. Яка команда надає змінній значення текстового типу?

- **a=float(ent1.get())**
- d=ent2.get()
- **b=float(ent3.get())**
- lab4.grid(row=1, column=2)
- btn1.grid(row=2, column=2)

**Тест до теми «Віджет Text – клас багаторядкового текстового поля»**

1. Який загальний синтаксис для створення багаторядкового текстового поля?

- **name = Text(window)**

- `Text(window) = name`
- `Text_name = (window)`
- `nameText = (window)`

2. Вкажіть властивості віджета `Text`

- **width**
- **height**
- **bg**
- **fg**
- `tkinter`
- `place`
- `Label`

3. Встановіть відповідність між функціями віджета `Text` та їх призначенням

|                                | <code>get</code><br>(початок,<br>кінець) | <code>insert</code><br>(позиція,<br>текст) | <code>delete</code><br>(початок,<br>кінець) |
|--------------------------------|------------------------------------------|--------------------------------------------|---------------------------------------------|
| Отримує фрагмент тексту з поля | +                                        |                                            |                                             |
| Вставляє текст в поле          |                                          | +                                          |                                             |
| Видаляє текст з поля           |                                          |                                            | +                                           |

4. Який загальний синтаксис для створення смуги прокручування?

- **`name = Scrollbar(window)`**
- `Scrollbar(window) = name`
- `Scrollbar_name = (window)`
- `nameScrollbar = (window)`

5. Щоб створити функціональний скроллер (тобто такий, що буде зв'язаний з віджетом та прокручуватиме його), потрібно виконати такі три кроки:

- **створити багаторядковий текстовий віджет;**
- **створити смугу прокручування і зв'язати її з текстовим віджетом;**

- **зв'язати текстовий віджет з панеллю прокрутки**
- створити кнопку
- створити текстовий напис (мітку)
- створити однорядковий текстовий віджет;

6. scrollbar = Scrollbar(root, command=textWidget.yview)

- **це команда створення смуги прокручування і зв'язка з віджетом textWidget**
- це команда видалення смуги прокручування і зв'язка з віджетом textWidget
- це команда створення текстового поля і зв'язка з віджетом textWidget
- це команда очистки текстового поля

7. root = Tk()

- **це команда створення головного вікна**
- це команда створення напису
- це команда створення кнопки
- це команда створення однорядкового текстового поля
- це команда створення багаторядкового текстового поля

8. txt = Text(root, width=30, height=5, bg="#ffffff", fg='#1E17E7') Таким рядком програми у текстовому полі буде налаштовано:

- **ширина текстового поля**
- **висота текстового поля**
- **колір фону текстового поля**
- **колір тексту текстового поля**
- колір рядка заголовка вікна програми
- координати розміщення текстового поля у вікні
- колір меж текстового поля

9. btn\_ins = Button(root, text="Вставити", command=insertText)

- це команда створює кнопку, що належить вікну **root**
- це команда створює кнопку, що належить вікну **text**
- це команда створює кнопку, що належить вікну **command**
- це команда створює кнопку, що належить вікну **insertText**
- це команда створює кнопку, що належить вікну **Button**
- це команда створює кнопку, що належить вікну **btn\_ins**

#### 10. `btn_del.pack(side=LEFT)`

- ця команда видалить кнопку `pack` зліва-направо
- ця команда видалить кнопку `pack` справа-наліво
- ця команда спрацьовує при натисненні ЛІВОЇ клавіші миші
- **ця команда може відображати кнопку у вікні програми**

### *Тест до теми «Radiobutton та Checkbutton – перемикачі та прапорці».*

1. Цей віджет потрібен користувачам для вибору кількох елементів у вікні
  - Radiobutton
  - **Checkbutton**
  - Label
  - Button
  - Entry
  - Text
  - Canvas
  - Scrollbar
2. Зв'язок встановлюється через загальну змінну, різні значення якої відповідають включенню різних перемикачів у групі. Це про...

- **Radiobutton**
- Checkbutton
- Label
- Button
- Entry
- Text
- Canvas
- Scrollbar

3. Вони потрібні, щоб отримувати відомості про стан прапорців. За значенням пов'язаної з Checkbutton \_\_\_\_\_ можна визначити, чи обрано прапорець чи ні, що в свою чергу вплине на хід виконання програми. Яке слово пропущено у реченні?

- **змінної**
- функції
- константи
- похідної
- абсциси
- ординати
- програми

4. Яка змінна визначає стан прапорця у даному прикладі?

```
check_b1 = Checkbutton(root, text="Я погоджуюся з умовами угоди", variable=var1, onvalue=1, offvalue=0)
```

- check\_b1
- Checkbutton
- root
- text
- variable
- **var1**
- onvalue
- offvalue

5. У якому вікні розміщено прапорець у даному прикладі?



```
check_b1 = Checkbutton(root, text="Я погоджуюся з умовами угоди", variable=var1, onvalue=1, offvalue=0)
```

- check\_b1
- Checkbutton
- **root**
- text
- variable
- var1
- onvalue
- offvalue

6. Яке ім'я об'єкта "прапорець" у даному прикладі?

```
check_b1 = Checkbutton(root, text="Я погоджуюся з умовами угоди", variable=var1, onvalue=1, offvalue=0)
```

- **check\_b1**
- Checkbutton
- root
- text
- variable
- var1
- onvalue
- offvalue

7. Якого значення набуде змінна при УВІМКНЕНОМУ прапорці у даному прикладі?

```
check_b1 = Checkbutton(root, text="Я погоджуюся з умовами угоди", variable=var1, onvalue=1, offvalue=0)
```

- check\_b1
- Checkbutton
- root
- text
- variable
- var1
- onvalue
- offvalue

- Я погоджуюся з умовами угоди
- 1
- 0

8. Якого значення набуде змінна при ВИМКНЕНОМУ прапорці у даному прикладі?

```
check_b1 = Checkbutton(root, text="Я погоджуюся з умовами угоди", variable=var1, onvalue=1, offvalue=0)
```

- check\_b1
- Checkbutton
- root
- text
- variable
- var1
- onvalue
- offvalue
- Я погоджуюся з умовами угоди
- 1
- 0

9. Tkinter не може використовувати будь-яку змінну для зберігання станів віджетів. Для цих цілей передбачені спеціальні класи-змінні пакета tkinter. Встановіть відповідність між назвою класу змінної та її описом.

|                                  | Boolean<br>Var | Int<br>Var | Double<br>Var | String<br>Var | Text<br>Var | Drib<br>Var | Tsile<br>Var |
|----------------------------------|----------------|------------|---------------|---------------|-------------|-------------|--------------|
| приймає булеві значення          | +              |            |               |               |             |             |              |
| приймає цілі значення            |                | +          |               |               |             |             |              |
| приймає дробові та цілі значення |                |            | +             |               |             |             |              |
| приймає рядкові значення         |                |            |               | +             |             |             |              |

## *Тест до теми «Віджет Canvas – клас полотна для малювання»*

1. Загальний синтаксис створення полотна:

- **name = Canvas(window)**
- name = Canva(window)
- Canvas = name(window)
- window\_Canvas = name()
- window=name(Canvas)
- Canva = name(window)

2. Під час розміщення геометричних примітивів вказуються їхні координати на полотні. Точкою відліку є ...

- **лівий верхній кут**
- правий верхній кут
- лівий нижній кут
- правий нижній кут
- центр полотна
- будь-яка точка полотна, яку задають в програмі може бути точкою відліку

3. У поданому нижче коді полотно має ім'я...

```
canv = Canvas(root, width=200, height=400)
```

- **canv**
- Canvas
- root
- width
- height

4. У поданому нижче коді використано клас об'єктів...

```
canv = Canvas(root, width=200, height=400)
```

- canv
- **Canvas**
- root
- width
- height

5. У поданому вище коді створене полотно має розміри...

```
canv = Canvas(root, width=200, height=400)
```

- **ширина - 200 пікселів, висота - 400 пікселів**
- висота - 200 пікселів, ширина - 400 пікселів
- ширина - 200 мм, висота - 400 мм
- висота - 200 мм, ширина - 400 мм
- висота - 200 дюймів, ширина - 400 дюймів
- ширина - 200 дюймів, висота - 400 дюймів

6. Таким кодом буде створено ...

```
c.create_line(50, 100, 200, 100,
 fill='red',
 width=5,
 dash=(10, 2),
 activefill='yellow',
 arrow=LAST,
 arrowshape="10 30 10")
```

- відрізок довжиною 50 пікселів
- відрізок довжиною 100 пікселів
- відрізок довжиною 200 пікселів
- відрізок довжиною 150 пікселів
- стрілка довжиною 50 пікселів
- стрілка довжиною 100 пікселів
- стрілка довжиною 200 пікселів
- **стрілка довжиною 150 пікселів**

7. Цим кодом буде створено лінію..

```
c.create_line(50, 100, 200, 100,
 fill='red',
 width=5,
 dash=(10, 2),
 activefill='yellow',
 arrow=LAST,
 arrowshape="10 30 10")
```

- суцільну ( \_\_\_\_\_ )
- пунктирну – лінію з крапочок ( ..... )
- **штрихову** ( - - - - - )
- штрих-пунктирну ( - · - · - · - · - · - · - · - · - )
- подвійну штрихову ( = = = = = )

8. Методом `create_rectangle` можна побудувати...

- **прямокутники**
- **квадрати**
- паралелограми
- трапеції
- ромби

9. Методом `create_polygon` можна побудувати

- **прямокутники**
- **квадрати**
- **паралелограми**
- **трапеції**
- **ромби**
- **трикутники**
- **п'ятикутники**
- **шестикутники**
- коло
- еліпс
- сферу

10. Методом `create_arc` можна зобразити

- **сектор**
- **сегмент**
- **дугу**
- трикутник
- зірку
- ламану лінію

11. Залежно від значення параметра style методу create\_arc можна побудувати

|         | за замовчуванням | CHORD | ARC |
|---------|------------------|-------|-----|
| сегмент |                  | +     |     |
| дугу    |                  |       | +   |

## Курс за вибором «Програмування на Java»

Укладач Попруга Т.О.

Рекомендації складено відповідно до програми «Програмування на Java» (авт. Іщеряков С.М., Двояк Г.П., Мішагіна Д=О.Д., Лист ІМЗО №22.1/12-Г-18 від 23.01.2019)

Посилання:

[https://drive.google.com/file/d/1oj1yLNGveIPVknRn\\_wOsvuShA8BS4vDv/view](https://drive.google.com/file/d/1oj1yLNGveIPVknRn_wOsvuShA8BS4vDv/view)

### Розподіл годин на вивчення курсу (1 семестр – 70 годин)

| Тема                                                                                                                                                                                                                            | Кількість годин |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| Тема 1: Вступ до програмування мовою Java <ul style="list-style-type: none"><li>Вступ до мови програмування Java</li><li>Встановлення та налаштування середовища розробки (IDE)</li></ul>                                       | 8               |
| Тема 2: Основи мови Java та середовища розробки <ul style="list-style-type: none"><li>Синтаксис мови Java: змінні, типи даних, оператори</li><li>Введення-виведення даних</li><li>Робота зі змінними та константами</li></ul>   | 8               |
| Тема 3: Управління потоком виконання програми <ul style="list-style-type: none"><li>Умовні оператори: if, if-else, switch</li><li>Цикли: for, while, do-while</li><li>Керування потоком виконання: break, continue</li></ul>    | 12              |
| Тема 4: Масиви та колекції <ul style="list-style-type: none"><li>Масиви: оголошення, ініціалізація, операції з масивами</li><li>Робота зі строками (String)</li><li>Колекції: ArrayList, LinkedList, HashSet, HashMap</li></ul> | 12              |

|                                                                                                                                                                                                                                                                                                                                                                     |    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Тема 5: Об'єктно-орієнтоване програмування:<br>основи <ul style="list-style-type: none"> <li>• Класи та об'єкти: оголошення, конструктори, методи</li> <li>• Інкапсуляція, змінні екземпляра, модифікатори доступу</li> <li>• Наслідування: класи-нащадки, ключове слово "extends"</li> <li>• Поліморфізм: перевизначення методів, динамічна зв'язування</li> </ul> | 16 |
| Тема 6: Об'єктно-орієнтоване програмування:<br>спадкування <ul style="list-style-type: none"> <li>• Багаторівневе спадкування класів</li> <li>• Абстрактні класи та методи</li> <li>• Інтерфейси: оголошення, реалізація, використання</li> </ul>                                                                                                                   | 14 |
| Разом                                                                                                                                                                                                                                                                                                                                                               | 70 |

### Список рекомендованої літератури для вчителя:

1. Вивчаємо Java у школі: навч. посіб. у 2 ч. : Ч. 1. Синтаксис мови / В.Д. Руденко, О.О. Жугастров. – Х. : Вид-во «Ранок», 2016. – 96 с.
2. Вивчаємо Java у школі: навч. посіб. у 2 ч. : Ч. 2. Класи, об'єкти, методи / В.Д. Руденко, О.О. Жугастров. – Х. : Вид-во «Ранок», 2016. – 96 с.
3. Руденко В.Д., Жугастров О.О. Перші кроки програмування мовою Java: навчальний посібник для учнів 10-11 класів. – Х.: РАНОК, 2018. 112 с.
4. Копитко М.Ф., Іванків К.С. Основи програмування мовою Java: Тексти лекцій. – Львів: Видавничий центр ЛНУ ім. Івана Франка, 2002.– 83 с. URL: <https://ami.lnu.edu.ua/wp-content/uploads/2017/05/Java.pdf>
5. Основи програмування (мовою Java): курс лекцій / О. В. Придатко, О. В. Хлевной, Н. Є. Бурак,. – Львів : ЛДУ



БЖД, 2019. – 180 с. URL: <https://books.ldubgd.edu.ua/index.php/ed/catalog/view/33/2/1/94-1>

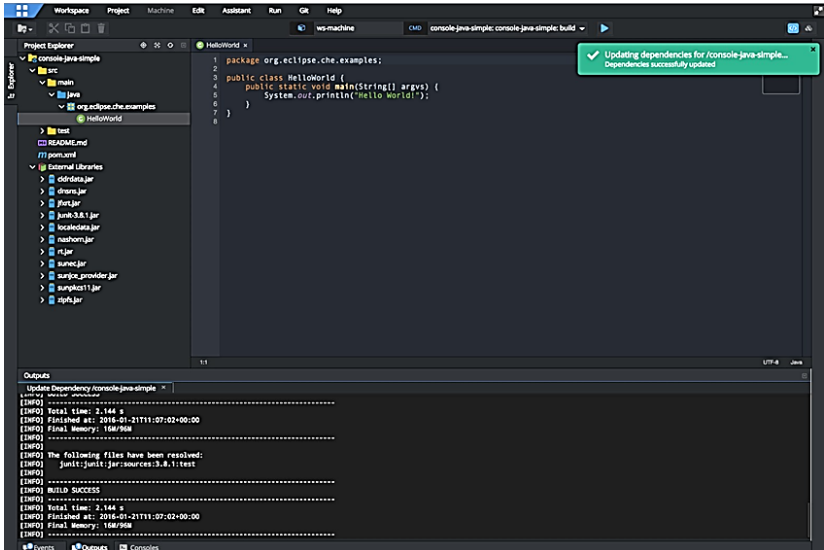
6. Чудна О. М. До викладання курсу за вибором «Програмування на Java». *НАУКОВИ ЗАПИСКИ МОЛОДИХ УЧЕНИХ*, №8 (2021). URL: <https://phm.cuspu.edu.ua/ojs/index.php/SNYS/article/view/1887/pdf>

### **Список рекомендованої літератури та електронних джерел для учнів:**

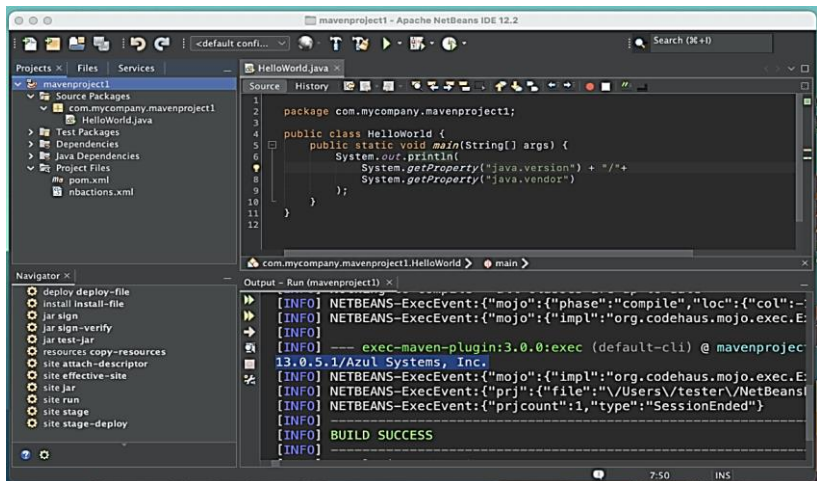
1. Вивчаємо Java у школі: навч. посіб. у 2 ч. : Ч. 1. Синтаксис мови / В.Д. Руденко, О.О. Жугастров. – Х. : Вид-во «Ранок», 2016. – 96 с.
2. Вивчаємо Java у школі: навч. посіб. у 2 ч. : Ч. 2. Класи, об'єкти, методи / В.Д. Руденко, О.О. Жугастров. – Х. : Вид-во «Ранок», 2016. – 96 с.
3. Руденко В.Д., Жугастров О.О. Перші кроки програмування мовою Java: навчальний посібник для учнів 10-11 класів. – Х.: РАНОК, 2018. 112 с.
4. W3schools UA (українською)
5. Yakov Fain. Java Programming for Kids, Parents and Grandparents, 2011 (є переклад російською: Файн Я. Программирование на JAVA для детей, родителей, бабушек и дедушек. – 231с. URL: [http://myflex.org/books/java4kids/JavaKid8x11\\_ru.pdf](http://myflex.org/books/java4kids/JavaKid8x11_ru.pdf))
6. Javarush - онлайн-курс навчання програмування на Java.

### **Програмне забезпечення:**

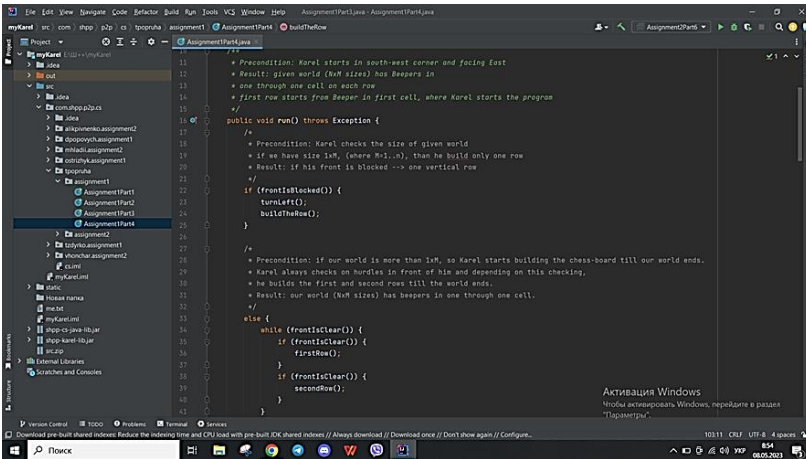
1. **Eclipse** (<https://www.eclipse.org/etrice/downloads/>) – є одним з найпопулярніших та потужних інтегрованих середовищ розробки для Java. Воно має широкі можливості, такі як редагування коду, компіляція, налагодження та керування проектами.



2. **NetBeans** (<https://netbeans.apache.org/>) – є відкритим інтегрованим середовищем, яке підтримує розробку на Java, а також інших мовах програмування. Воно має інтуїтивний інтерфейс та велику кількість інструментів для розробки, таких як редактор коду, налагоджувач, керування проектами тощо.

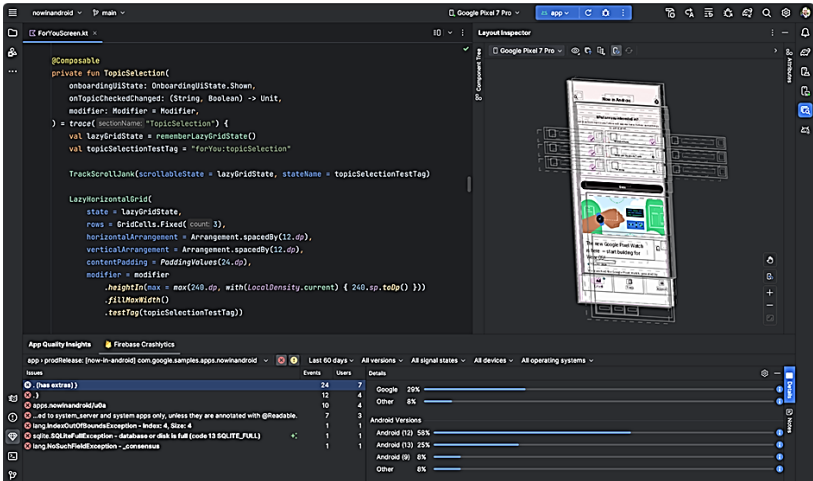


3. **IntelliJ IDEA** (<https://www.jetbrains.com/idea/download/#section=windows>) – Community Edition є безкоштовною версією інтегрованого середовища розробки, яке надає велику кількість функцій та інструментів для розробки на Java. Воно підтримує автоматичне завершення коду, рефакторинг, налагодження та багато іншого



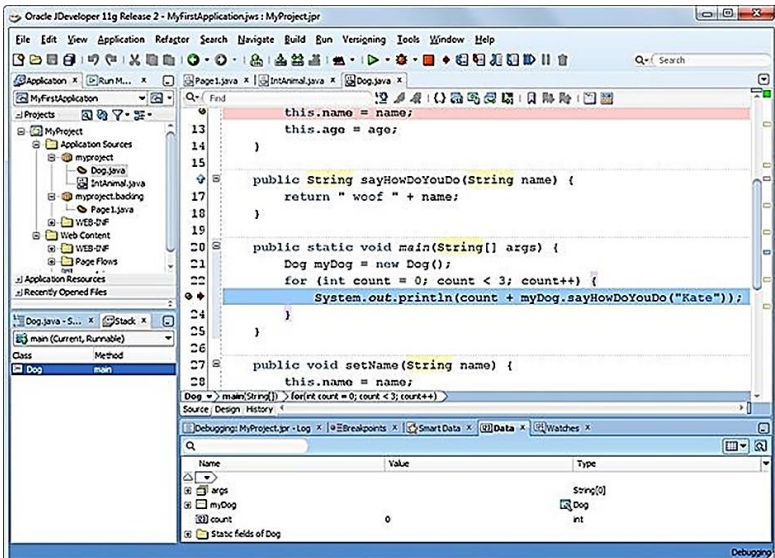
4. **Android Studio** (<https://developer.android.com/studio>) – Android Studio підтримує розробку додатків на мові програмування Java для платформи Android. Ви можете використовувати Android Studio для розробки нативних Android-додатків, використовуючи Java як основну мову програмування.

При розробці додатків на Java в Android Studio ви можете використовувати всі потужні функції та інструменти, які надаються цим середовищем розробки. Ви зможете редагувати та компілювати Java-код, створювати і налаштовувати інтерфейс користувача, виконувати налагодження, тестувати та розгортати додатки на пристроях або емуляторах Android.

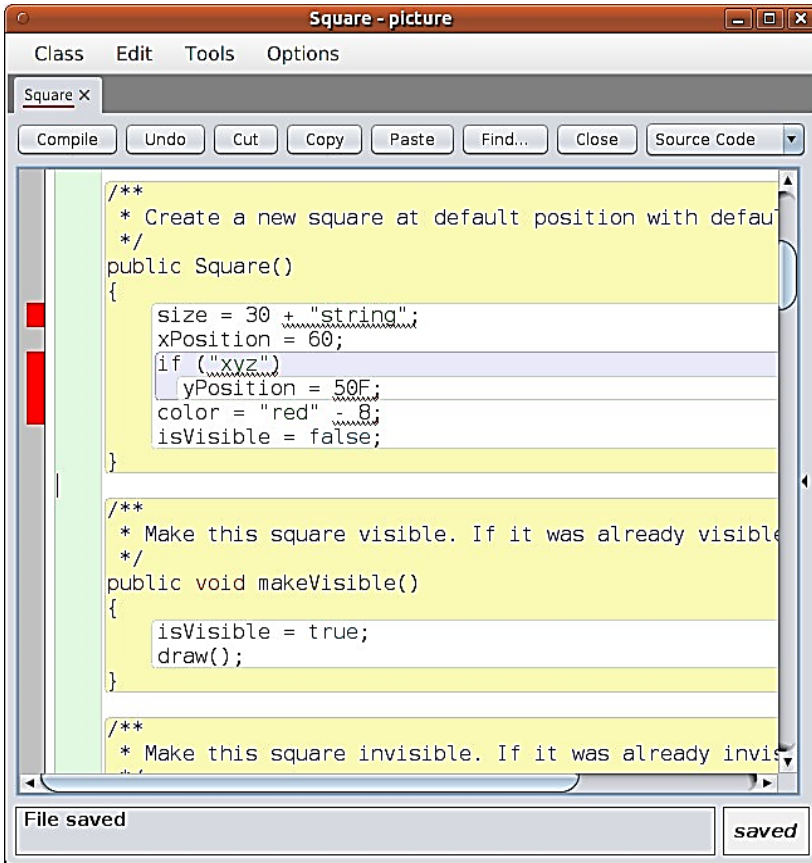


5. **JDeveloper** ([https://www.oracle.com/cis/application-development/technologies/jdevel\\_oper.html](https://www.oracle.com/cis/application-development/technologies/jdevel_oper.html)) – інтегроване середовище розробки (IDE) від компанії Oracle, призначене для розробки програмного забезпечення різними мовами програмування, включаючи Java.

JDeveloper надає розширені можливості для розробки, налагодження, тестування та розгортання Java-додатків.



6. **BlueJ** (<https://www.bluej.org/>) – є легким і простим у використанні інтегрованим середовищем розробки, спеціально призначеним для вивчення програмування на Java. Воно надає зручні інструменти для редагування коду, виконання та налагодження програм.



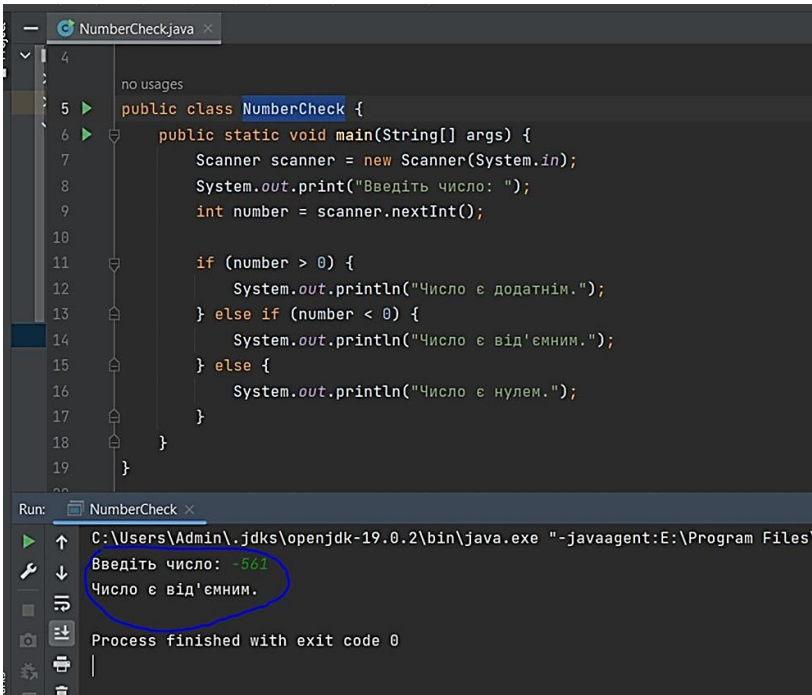
## Приклади практичних завдань

### Завдання до теми «Оператори розгалуження»

*Завдання 1 (Легкий рівень):*

**Задача:** Напишіть програму, яка приймає введене число і перевіряє, чи є воно додатнім, від'ємним чи нулем.

## Розв'язання:



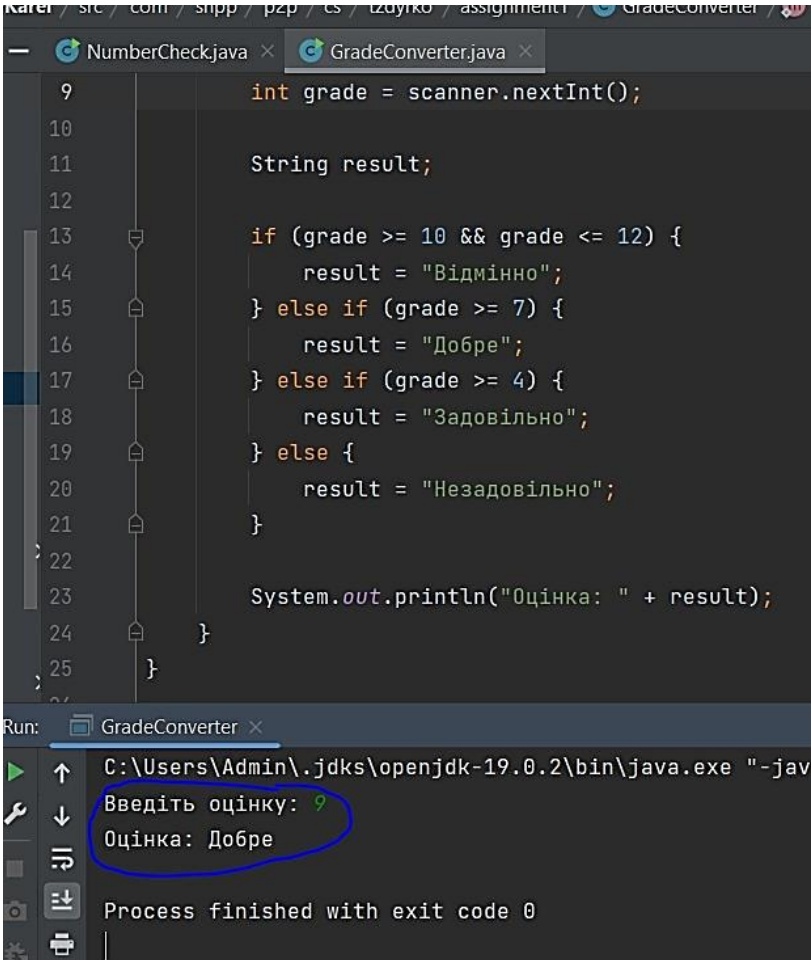
```
4
no usages
5 public class NumberCheck {
6 public static void main(String[] args) {
7 Scanner scanner = new Scanner(System.in);
8 System.out.print("Введіть число: ");
9 int number = scanner.nextInt();
10
11 if (number > 0) {
12 System.out.println("Число є додатнім.");
13 } else if (number < 0) {
14 System.out.println("Число є від'ємним.");
15 } else {
16 System.out.println("Число є нулем.");
17 }
18 }
19 }
20 }
Run: NumberCheck ×
C:\Users\Admin\.jdk\openjdk-19.0.2\bin\java.exe "-javaagent:E:\Program Files
Введіть число: -561
Число є від'ємним.
Process finished with exit code 0
```

**Пояснення:** В даному розв'язанні використовується оператор розгалуження if-else. За допомогою умовних перевірок визначається, чи є число додатнім, від'ємним чи нулем, і виводиться відповідне повідомлення.

*Завдання 2 (Середній рівень):*

**Задача:** Напишіть програму, яка приймає введену оцінку студента (ціле число) і виводить відповідну оцінку в текстовому форматі ("Відмінно", "Добре", "Задовільно" або "Незадовільно").

## Розв'язання:



```
9 int grade = scanner.nextInt();
10
11 String result;
12
13 if (grade >= 10 && grade <= 12) {
14 result = "Відмінно";
15 } else if (grade >= 7) {
16 result = "Добре";
17 } else if (grade >= 4) {
18 result = "Задовільно";
19 } else {
20 result = "Незадовільно";
21 }
22
23 System.out.println("Оцінка: " + result);
24 }
25 }
```

Run: GradeConverter ×

```
C:\Users\Admin\.jdk\openjdk-19.0.2\bin\java.exe "-jav
Введіть оцінку: 9
Оцінка: Добре
Process finished with exit code 0
```

**Пояснення:** В цьому розв'язанні застосовується ланцюжок умовних перевірок if-else if. Залежно від значення оцінки студента, встановлюється відповідна текстова оцінка в змінну result, яка потім виводиться на екран.

*Завдання 3 (Високий рівень):*

**Задача:** Напишіть програму, яка знаходить найбільше з трьох введених чисел.

## Розв'язання:

```
NumberCheck.java x GradeConverter.java x LargestNumberFinder.java x
3 import java.util.Scanner;
4
5 no usages
6 public class LargestNumberFinder {
7 public static void main(String[] args) {
8 Scanner scanner = new Scanner(System.in);
9 System.out.print("Введіть перше число: ");
10 int number1 = scanner.nextInt();
11 System.out.print("Введіть друге число: ");
12 int number2 = scanner.nextInt();
13 System.out.print("Введіть третє число: ");
14 int number3 = scanner.nextInt();
15
16 int largestNumber;
17
18 if (number1 >= number2 && number1 >= number3) {
19 largestNumber = number1;
20 } else if (number2 >= number1 && number2 >= number3) {
21 largestNumber = number2;
22 } else {
23 largestNumber = number3;
24 }
25 System.out.println("Найбільше число: " + largestNumber);
26 }
27 }
```

```
GradeConverter x LargestNumberFinder x
C:\Users\Admin\.jdk\openjdk-19.0.2\bin\java.exe "-javaagent:E:\Program
Введіть перше число: -12
Введіть друге число: 10
Введіть третє число: 0
Найбільше число: 10
```

**Пояснення:** У даному завданні, для знаходження найбільшого числа з трьох введених, застосовується ланцюжок умовних перевірок if-else if.

Програма порівнює перше число (number1) з другим числом (number2) та третім числом (number3).



Якщо  $number1 \leq number2$  і  $number1 \leq number3$ , то  $number1$  є найбільшим числом. У цьому випадку, значення  $number1$  надається змінній  $largestNumber$ .

Якщо перша умова не виконується, то перевіряється умова  $number2 \leq number1$  і  $number2 \leq number3$ .

Якщо ця умова виконується, то  $number2$  є найбільшим числом, і його значення надається  $largestNumber$ .

Якщо жодна з перших двох умов не виконується, це означає, що найбільшим числом є  $number3$ , і його значення надається  $largestNumber$ .

Нарешті, найбільше число ( $largestNumber$ ) виводиться на екран.

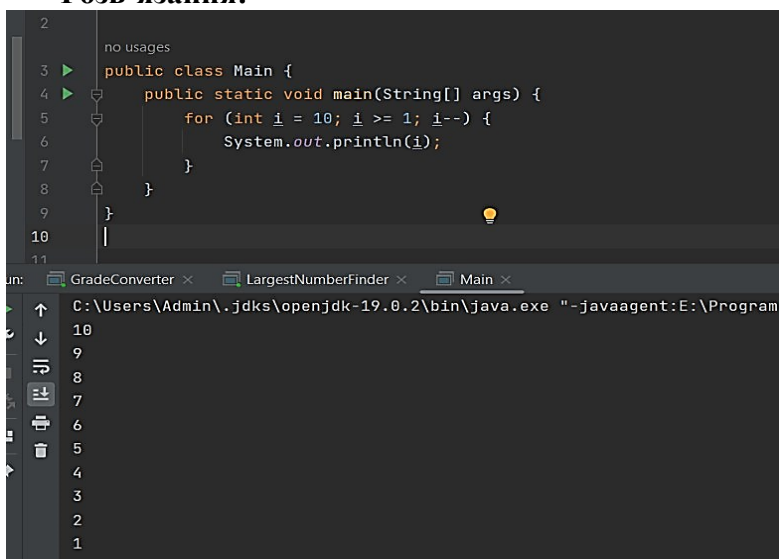
Цей розв'язок дозволяє знайти найбільше число з трьох введених і забезпечує точність порівнянь між ними.

## Завдання до теми «Оператори цикла»

Завдання 1 (легкий рівень):

**Задача:** Напишіть програму, яка виводить на екран перші 10 натуральних чисел у зворотному порядку.

**Розв'язання:**



```
2
no usages
3 public class Main {
4 public static void main(String[] args) {
5 for (int i = 10; i >= 1; i--) {
6 System.out.println(i);
7 }
8 }
9 }
10
11
```

Ln: GradeConverter x LargestNumberFinder x Main x

C:\Users\Admin\jdk\openjdk-19.0.2\bin\java.exe "-javaagent:E:\Program

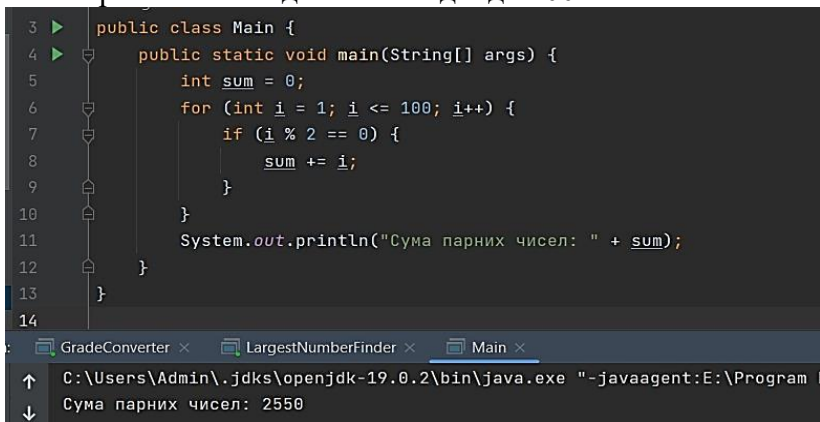
10  
9  
8  
7  
6  
5  
4  
3  
2  
1

### Пояснення:

У даному завданні ми використовуємо цикл `for`, який починається зі значення 10 і продовжує до 1 з кроком  $-1$ . На кожній ітерації циклу виводиться поточне значення змінної `i` на екран.

### Завдання 2 (середній рівень):

**Завдання:** Напишіть програму, яка обчислює суму всіх парних чисел в діапазоні від 1 до 100.



```
3 public class Main {
4 public static void main(String[] args) {
5 int sum = 0;
6 for (int i = 1; i <= 100; i++) {
7 if (i % 2 == 0) {
8 sum += i;
9 }
10 }
11 System.out.println("Сума парних чисел: " + sum);
12 }
13 }
14
```

↑ C:\Users\Admin\jdk8\openjdk-19.0.2\bin\java.exe -javaagent:E:\Program  
↓ Сума парних чисел: 2550

**Пояснення:** У даному завданні ми використовуємо цикл `for`, який проходить усі числа від 1 до 100. Використовуючи оператор `%` (оператор залишку від ділення), ми перевіряємо, чи є поточне число парним. Якщо число є парним, то додаємо його до змінної `sum`. На кінці програми виводимо суму парних чисел на екран.

### Завдання 3 (середній ++ рівень):

**Завдання:** Напишіть програму, яка виводить на екран таблицю множення для числа 7 в діапазоні від 1 до 10.

```
3 > public class Main {
4 > public static void main(String[] args) {
5 int number = 7;
6 for (int i = 1; i <= 10; i++) {
7 int result = number * i;
8 System.out.println(number + " * " + i + " = " + result);
9 }
10 }
11 }
12 |
13
```

GradeConverter x LargestNumberFinder x Main x

```
C:\Users\Admin\.jdk\openjdk-19.0.2\bin\java.exe "-javaagent:E:\Program Files\Intel
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
7 * 10 = 70

Process finished with exit code 0
```

**Пояснення:** Ця програма демонструє використання циклу `for` для створення таблиці множення для заданого числа. Кожна ітерація циклу обчислює добуток числа 7 і поточного числа `i`, після чого результат виводиться на екран у зрозумілому форматі.

*Завдання 4 (мега високий рівень):*

**Завдання:** Напишіть програму, яка знаходить суму цифр усіх чисел у заданому діапазоні від `start` до `end`, включно. Потім програма виводить суму цифр на екран.

```

3 public class Main {
4 public static void main(String[] args) {
5 int start = 100;
6 int end = 200;
7 int sumOfDigits = 0;
8
9 for (int i = start; i <= end; i++) {
10 int number = i;
11 int digitSum = 0;
12
13 while (number != 0) {
14 int digit = number % 10;
15 digitSum += digit;
16 number /= 10;
17 }
18
19 sumOfDigits += digitSum;
20 }
21 System.out.println("Сума цифр усіх чисел в діапазоні [" + start + ", " + end + "]: " + sumOfDigits);
22 }
23 }

```

### Пояснення:

У даному завданні використовуються два цикли: зовнішній цикл `for`, що проходить через всі числа від `start` до `end`, і внутрішній цикл `while`, який обчислює суму цифр кожного числа.

В кожній ітерації зовнішнього циклу береться поточне число `i`. Внутрішній цикл `while` використовується для розбиття числа на окремі цифри. З кожної цифри вираховується остання цифра (`digit`), і до загальної суми `digitSum` додається ця цифра. Далі число `number` ділиться на 10, щоб перейти до наступної цифри. Цей процес повторюється, поки всі цифри числа не будуть оброблені.

Сума цифр кожного числа `digitSum` додається до загальної суми `sumOfDigits`. На кінці програми виводиться сума цифр усіх чисел у заданому діапазоні на екран.

Це завдання вимагає більшої складності і включає в себе вкладені цикли та розбиття чисел на окремі цифри. Воно дає можливість вдосконалити навички роботи з циклами та операціями над числами.

Наші автори:  
Величко Марія Олексіївна  
Настусенко Софія Дмитрівна  
Теренчин Богдан Валерійович  
Левицький Ярослав В'ячеславович  
Попруга Тетяна Олегівна

Підписано до друку 20.09.2023. Формат 60×84 <sup>1</sup>/<sub>16</sub>.  
Папір офсет. Цифровий друк. Ум.друк. арк. 10,46. Обл. вид. арк. 3,49.  
Видруковано: ФОП Піскова М. А.  
Свідоцтво про державну реєстрацію  
№2444000000027816 від 18.08.2016