

**Основи  
структурного (алгоритмічного)  
програмування  
мовою Java**

*Навчальна програма спецкурсу*

*для слухачів курсів підвищення кваліфікації  
(вчителі та методисти інформатики, математики, фізики)  
дистанційної форми навчання*

## ПОЯСНЮВАЛЬНА ЗАПИСКА

Зміст навчальної програми курсу “Основи структурного (алгоритмічного) програмування мовою Java” наближено, з одного боку, до бізнес-вимог сучасного ІТ-ринку, що повинно прискорити процес самореалізації школяра в суспільстві. З другого боку, застосування в якості основного навчального джерела комплексу відеоматеріалів “Програмування на Java”, що має гриф “Схвалено для використання у загальноосвітніх навчальних закладах” (лист ІМЗО № 22.1/12-Г-558 від 10.07.2020), дозволяє повноцінно реалізувати в освітньому процесі принцип “перевернутого кабінету” (blended learning), найбільш ефективного для дистанційного навчання.

**Категорія слухачів:** вчителі інформатики, математики, фізики загальноосвітніх навчальних закладів, працівники методичних служб, науково-педагогічні працівники закладів післядипломної педагогічної освіти.

**Мета курсу:** підвищення кваліфікації вчителів інформатики, математики, фізики та методистів до рівня, мінімально необхідного для створення власних простих програмних продуктів на основі сучасних парадигм та технологій.

**Завданнями курсу є:**

- формування у вчителів інформатики знань та навичок, мінімально необхідних для підняття рівня проведення уроків змістової лінії “Алгоритмізація та програмування” до вимог сучасного ІТ-ринку;

- надання вчителям математики і фізики знань та навичок з програмування, необхідних для імплементування елементів ІТ у процес вивчення математики та фізики із формуванням основ для реальної STEM-освіти, а також можливого зняття напруги із кадровою нестачею щодо вчителів інформатики в багатьох загальноосвітніх навчальних закладах;

- збільшення мотивації учнів до вивчення інформатики шляхом наближення навчального матеріалу до вимог сучасного ІТ-ринку.

**Основні ідеї програми:**

1. Наповнення змісту програми сучасним змістом, здатним додатково мотивувати учнів до вивчення ІТ.

2. Комплексне поєднання теоретичної та практичної складової в онлайн заняттях (вебінарах).

3. Насиченість навчальних тем тестовими завданнями, що активують самостійну роботу слухачів.

**Зміст та навчально-тематичне планування програми представлено комплексом модулів:**

Модуль №1 "Основи структурного програмування" призначений для формування у вчителя ґрунтовних знань та навичок класичного програмування, пов'язаних із побудовою типових алгоритмів – розгалужених, циклічних, вкладених циклічних, а також різноманітних алгоритмів обробки масивів.

Модуль № 2 "Вступ до об'єктно-орієнтованого програмування " формує у вчителя первинні навички використання можливостей об'єктної парадигми в процесі створення програм. У модулі актуалізовано незначну частину елементів та технологій об'єктно-орієнтованого програмування, у зв'язку із чим термін "об'єктно-орієнтоване програмування" не включено до назви курсу. Проте, обсяг навчального матеріалу є достатнім для свідомого написання проектів із використанням ресурсів двох основних стандартних класів мови Java – String та ArrayList.

**Після вивчення курсу слухачі:**

*Повинні знати:*

1. Основи алгоритмізації обчислювальних процесів.
2. Особливості використання в алгоритмах даних різних типів.
3. Основні поняття об'єктної парадигми програмування.

*Повинні уміти:*

1. Використовувати сучасні інтегровані середовища розробника для написання програм.
2. Застосовувати найбільш ефективні типи даних в програмах.
3. Складати типові розгалужені алгоритми для написання циклічних програм, в т.ч. із вкладеними циклами, а також для програм обробки масивів.
4. Використовувати методи стандартних класів – String, ArrayList.

**Стратегія навчання:**

Реалізація мети і завдань програми курсу здійснюється в умовах дистанційного підвищення кваліфікації методистів, вчителів інформатики, математики та фізики навчальних закладів шляхом:

- опанування її змісту, відповідних умінь і компетентностей у процесі онлайн занять – вебінарів;
- самостійних навчальних практикумів, спрямованих на закріплення й подальший розвиток умінь і компетентностей за рахунок самостійного виконання тестових та проектних завдань;
- зворотного зв'язку у вигляді менторської підтримки слухачів.

**Обсяг програми:**

Спецкурс розрахований на 16 академічних годин онлайн занять (вебінарів) із аналізом (review) програмних кодів завдань навчального практикуму та 14 годин самостійної роботи по опрацюванню відеоматеріалів та виконанню завдань навчального практикуму (разом – 30 годин, 1 кредит ЄКТС).

**Компетентність, що вдосконалюватиметься:** інформатична.

**Розробник програми:**

*Ищераков Сергій Михайлович*, кандидат технічних наук, доцент ІТ-кафедр Державного університету телекомунікацій (м.Київ), НТУУ “Київський політехнічний інститут імені Ігоря Сікорського”, Прикарпатського Національного університету імені Василя Стефаника, ПВНЗ “ІТ Степ Університет”(м.Львів), ініціатор проєкту “Школяр-програміст” з успішним досвідом організації Java-тренінгів різного рівня з 2015-го року для учнів, вчителів та студентів навчальних закладів України, підготував кілька десятків учнів, сертифікованих корпорацією Oracle (США) як Java-програмістів, володар відзнаки “За видатний внесок у розвиток ІТ-освіти в Україні” на конкурсі вітчизняного ІТ-бізнесу IT Awards 2017.

**Програму затверджено:**

директором ТОВ “Інновейшн Девелопмент Фаундейшн” Тетяною Рахуба 22 грудня 2021 року.

# НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

## Модуль 1. Основи структурного (алгоритмічного) програмування

**Тема 1. Робота з Eclipse. Перша програма. Поняття клас, пакет, проєкт. Коментарі. Оголошення та ініціалізація змінних. Вказівка присвоювання. Ідентифікатори. Ключові слова. Типи даних *String*, *int*, *double*. Арифметичні операції. Створення лінійних програм з використанням змінних. Перетворення (*casting*) типів *int* та *double*. Конкатенація *String*.**

Урок 1.1. Перша програма.

Урок 1.2. Змінні. Типи змінних. Арифметичні операції. Кастинг.

Урок 1.3. Конкатенація.\*

**Тема 2. Двійкове кодування.**

Урок 2.1. Представлення натуральних чисел в двійковій системі числення.

Урок 2.2. Представлення натуральних чисел в шістнадцятковій та вісімковій системах числення.

Урок 2.3. Представлення від'ємних чисел в двійковій системі числення.\*

**Тема 3. Логічні вирази та змінні й операції над ними. Умовні оператори (коротка та повна форма). Складені умови.**

Урок 3.1. Операції порівняння. Логічні змінні.

Урок 3.2. Оператори розгалуження if-else.

Урок 3.3. Вкладені оператори розгалуження if\_else

Урок 3.4. Логічні функції NOT, AND, OR.

**Тема 4. Оператори циклу. Вкладені цикли.**

Урок 4.1. Циклічні процедури. Оператор for.

Урок 4.2. Цикл for із внутрішніми операторами if.

Урок 4.3. Оператор циклу while.

Урок 4.4. Вкладені цикли. Побудова крапкових трикутників.

**Тема 5. Поняття одновимірного масиву. Виведення значень елементів масиву. Алгоритми опрацювання масивів.**

Урок 5.1. Поняття масиву. Виведення елементів масиву.

Урок 5.2. Сума елементів масиву цілих чисел.

Урок 5.3. Пошук максимуму в масиві цілих чисел.

Урок 5.4. Попарне додавання елементів масивів цілих чисел.

Урок 5.5. Конкатенація масивів.\*

Урок 5.6. Поділ масиву.\*

**Тема 6. Розширені оператори структурного програмування в Java.\***

Урок 6.1. Оператори з присвоєнням.\*

Урок 6.2. Оператори інкрементування-декрементування.\*

Урок 6.3. Тернарний оператор.\*

Урок 6.4. Логічна функція XOR.\*

Урок 6.5. Short-circuit логічні функції AND та OR.\*

Урок 6.6. Оператор switch.\*

Урок 6.7. Оператори continue та break.\*

Урок 6.8. Оператор forEach.\*

xxx.\* - тут і нижче підкресленням із зірочкою виділено теми підвищеної складності

## Модуль 2. Вступ до об'єктного програмування

**Тема 1. Ознайомлення із *Java API*, пакетом *java.lang*, класом *String* та методами *length*, *toLowerCase*, *toUpperCase*, *charAt*, *endsWith*, *startsWith* класу *String*. Ознайомлення із поняттям перезавантаження методів та методами *indexOf*, *replace*, *split*, *substring* класу *String*.**

Урок 1.1. API класу *String*. Методи *length*, *charAt*, *endsWith*.

Урок 1.2. Метод *indexOf*. Перезавантажені (*overload*) методи.

Урок 1.3. Методи *replace*, *substring*, *toUpperCase*, *split*.

**Тема 2. Ознайомлення із поняттям імпорт, ключовим словом *static* та класом *java.util.Arrays*. Методи *toString*, *sort*, *binarySearch* класу *Arrays*.**

Урок 2.1. Клас *Arrays*. Метод *toString*. Поняття *static* та *import*.

Урок 2.2. Методи *sort*, *binarySearch* класу *Arrays*.

**Тема 3. Клас *java.util.ArrayList*. Методи *size*, *add*, *get*, *indexOf* класу *ArrayList*. Обробка масивів цілих чисел із утворенням колекцій *ArrayList*.**

Урок 3.1. Клас *ArrayList*.

**Тема 4. Поняття інкапсуляції, структура класу. Архітектурний шаблон *MODEL-VIEW-CONTROLLER (MVC)* для побудови власних класів. Оголошення та ініціалізація полів. Оголошення методів.\***

Урок 4.1. Інкапсуляція. Шаблон MVC. Поля класу *Model*.\*

Урок 4.2. Створення методів в класах *Controller* та *View*.\*

**Тема 5. Оголошення та створення об'єктів. Оператор *new*. Літерал *null*. Приклад побудови проєкту *Calculator*. Ознайомлення із конструкторами, передаванням в метод посилань на об'єкти.\***

Урок 5.1. Зв'язування класів. Класи та об'єкти. Створення об'єктів. Оператор *new*.\*

Урок 5.2. Розміщення класів та об'єктів в пам'яті. Літерал *null*. Ініціалізація полів об'єктів.\*

Урок 5.3. Ознайомлення із конструкторами класів та передаванням в метод посилань на об'єкти.\*

**Тема 6. Виведення крапкових геометричних фігур за вимогами шаблону MVC.\***

# Модуль 1. Основи структурного (алгоритмічного) програмування

## ТЕМА 1

**Робота з Eclipse. Перша програма. Ознайомлення із поняттями клас, пакет, проєкт. Коментарі. Оголошення змінних. Ідентифікатори. Ключові слова. Типи даних *String*, *int*. Арифметичні операції. Тип даних *double*. Перетворення (*casting*) типів *int* та *double*. Конкатенація *String*.**

**Урок 1.1.** Робота з Eclipse. Перша програма. Ознайомлення із поняттями клас, пакет, проєкт. Коментарі.

**Відео:** Перша програма.

**Література основна:** [Руденко] – с.12...16 (розділ 1.3. Загальні відомості про мову Java), с.23 (підрозділ “Коментарі”).

**Література додаткова:** [Шилдт] – с.64 - 65 (розділ “Первый пример простой программы”), 66...69 (розділ “Подробный анализ первого примера программы”), (англ. – р.75, 76...79).

**Урок 1.2.** Оголошення змінних. Ідентифікатори. Ключові слова. Типи даних *String*, *int*. Арифметичні операції. Тип даних *double*. Перетворення (*casting*) типів *int* та *double*.

**Відео:** Змінні. Типи змінних. Арифметичні операції. Кастинг.

**Література основна:** [Руденко] – с.22...32 (розділи 2.1 Базові компоненти мови Java, 2.2 Прості типи даних і змінні, 2.3 Арифметичні оператори й арифметичні вирази, 2.4 Перевірка і перетворення типів).

**Література додаткова\*:** [Шилдт] – с.69...71 (розділ “Второй пример короткой программы”), с.75...78 (від розділу “Вопросы лексики” включно з розділом “Ключевые слова Java”), с.79...84 (включно з розділом “Тип double”), с.87...89 (розділи “Целочисленные литералы”, “Литералы с плавающей точкой”), с.90-91 (розділи “Переменные”, “Объявление переменных”), (англ. – р.79...81, 87...90, 92...96, 100...102, 103 - 104).

**Завдання:** [Руденко] – вибрані завдання для самостійного виконання розділів 2.1...2.4.

### Урок 1.3. Конкатенація String.\*

**Відео:** Конкатенація.

**Література:** [Шилдт] – с.533...534 (розділи “Сцепление строк”, “Сцепление символьных строк с другими типами данных”), (англ. – р.649...651).

**Тестові завдання:** [Ganesh] – Mock Test 1: № 52, 53 (р.535, answers and explanations– р.563).

## ТЕМА 2

**Представлення натуральних чисел у двійковій, шістнадцятковій та вісімковій системах числення. Представлення від'ємних чисел у двійковій системі числення.**

**Урок 2.1.** Представлення натуральних чисел в двійковій системі числення.

**Відео:** Представлення натуральних чисел в двійковій системі числення.

**Література:** [Щеряков] – с.6...10.

**Завдання** (перевірку реалізувати на Eclipse):

Дано беззнакові числа  $K, L$ , що є представленими в різних системах числення:

K (dec)	L (bin)
3009	0011 0110 0000
3555	0011 1101 1101
2996	0011 1100 1111
4050	0011 1011 1011
3120	0010 0111 1001

Перевести числа  $K, L$  з однієї системи числення до іншої:  $K$  (dec) = ... (bin),  
 $L$  (bin) = ... (dec).

**Урок 2.2.** Представлення натуральних чисел в шістнадцятковій та вісімковій системах числення.

**Відео:** Представлення натуральних чисел в шістнадцятковій та вісімковій системах числення.

**Література:** [Щеряков] – с.10...12.

**Завдання** (перевірку реалізувати на Eclipse):

Дано беззнакові числа  $K, M$ , що є представленими в різних системах числення:

K (dec)	M (hex)
3009	2AB
3555	4BA
2996	27E
4050	35B
3120	2B7

1. Перевести числа  $K, M$  з однієї системи числення до іншої із проміжним перетворенням у двійкову форму:

$$K \text{ (dec)} = \dots \text{ (bin)} = \dots \text{ (hex)},$$

$$M \text{ (hex)} = \dots \text{ (bin)} = \dots \text{ (dec)}.$$

2. В двійкових кодах виконати додавання  $K+M$ .



## Урок 2.3. Представлення від'ємних чисел в двійковій системі числення.\*

**Відео:** Представлення від'ємних чисел в двійковій системі числення.

**Література:** [Щеряков] – ст.52...57.

**Завдання:** представити у двійковому коді із розрядністю 32 біт наступні цілі від'ємні числа:

-10314      -22      -12345678

## ТЕМА 3

**Операції порівняння. Тип даних *boolean*. Оператори *if* та *if-else*. Логічні функції *NOT*, *AND*, *OR*.**

**Урок 3.1.** Операції порівняння. Тип даних *boolean*.

**Відео:** Операції порівняння. Логічні змінні.

**Література основна:** [Руденко] – с.32...33 (підрозділ “Оператори порівняння”)

**Література додаткова:** [Шилдт] – с.85...86 (розділ “Логические значения”), с.89 (розділ “Логические литералы”), с.124 (розділ “Операции отношения”), (англ. – р.98, 102, 141).

**Завдання:** [Руденко] – вибрані завдання для самостійного виконання розділу 2.6.

**Урок 3.2.** Оператори *if* та *if-else*.

**Відео:** Оператори розгалуження *if-else*.

**Література основна:** [Руденко] – с.40...44 (розділ 3.1.Програмування алгоритмів із розгалуженою структурою).

**Література додаткова:** [Шилдт] – с.131...133 (розділ “Условный оператор if”), (англ.– р.150...152),  
[Хорстманн] – с. 93...95 (англ. – р.140...142).

**Завдання:** [Руденко] – вибрані завдання для самостійного виконання розділу 3.1.

**Тестове завдання\*:** [Raposa] - ch.1: № 19.

**Урок 3.3.** Вкладені оператори *if-else*.

**Відео:** Вкладені оператори розгалуження *if\_else*

**Література основна:** [Руденко] – с.44...46 (розділ 3.2. Вкладені оператори умовного переходу”).

**Література додаткова:** [Шилдт] – с. 133...135 (розділи “Вложенные условные операторы if”, “Конструкция if-else-if”), (англ. – р.152...154),  
[Хорстманн] – с. 96...97 (англ. – р.143...145).

**Завдання:** [Руденко] – вибрані завдання для самостійного виконання розділу 3.2.

**Тестові завдання\*:** [Heller] - ch.5: № 4,  
[Mughal] - ch.6: № 6.1.

### Урок 3.4. Логічні функції *NOT*, *AND*, *OR*.

**Відео:** Логічні функції *NOT*, *AND*, *OR*.

**Література основна:** [Руденко] – с.33...34 (підрозділи “Логічні оператори”, “Логічні вирази”).

**Література додаткова:** [Шилдт] – с.125...126 (розділ “Логические операции”), (англ. – р.142, 143).

**Завдання:**

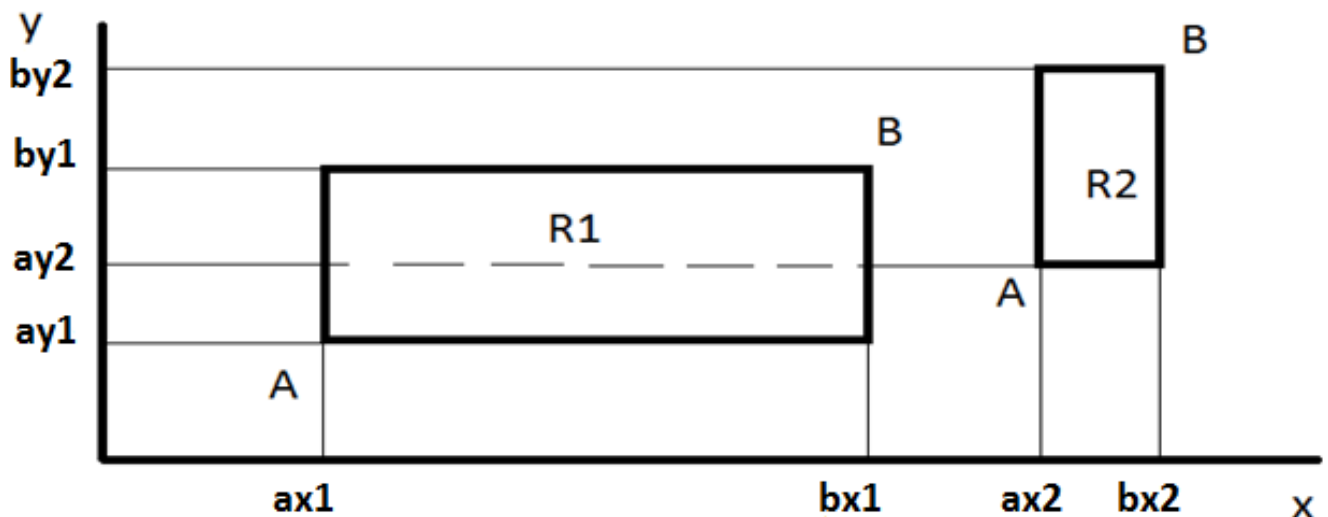
1.[Руденко] – вибрані завдання для самостійного виконання розділу 2.5.

2.Проектне завдання (детальне роз’яснення – у відео “Прямокутники”):

За допомогою операторів розгалуження та логічних функцій *AND*, *OR* визначити тип перетину двох прямокутників *R1* та *R2* із сторонами, що є паралельними вісям координат. Прямокутники задано координатами двох діагональних вершин *A* та *B*:

цілі (int) змінні *ax1*, *ay1*, *bx1*, *by1* – для прямокутника *R1*,

цілі (int) змінні *ax2*, *ay2*, *bx2*, *by2* – для прямокутника *R2*:



Для зменшення громіздкості коду припускаємо, що точка *A* має бути завжди нижче та лівіше точки *B*.

1. Написати Java-код для визначення двох з чотирьох можливих варіантів перетину:

1.1 обов’язковий варіант – перетину немає,

1.2 один з трьох можливих варіантів перетину за вибором слухача:

- перетином є крапка,
- перетином є вертикальна лінія,
- перетином є горизонтальна лінія.

2\*. Написати Java-код для визначення всіх чотирьох вищенаведених варіантів перетину, а також п’ятого варіанту, коли перетином є прямокутник.

РЕКОМЕНДОВАНО:

- обрати саме вищенаведену послідовність варіантів,

- кожний фрагмент коду для визначення певного типу перетину оформити у вигляді автономного відокремленого блоку операторів if-else.\*

## ТЕМА 4

**Цикли. Оператор *for*. Область видимості змінної. Вкладені цикли. Оператори *while* та *do-while*. Виведення крапкових трикутників.**

**Урок 4.1.** Цикли. Оператор *for*. Область видимості змінної.

**Відео:** Циклічні процедури. Оператор *for*.

**Література основна:** [Руденко] – с.53...57 (підрозділ “Цикли з лічильником”).

**Література додаткова\*:** [Шилдт] – с.92...95 (розділ “Область видимости и срок действия переменных”), с.145...149 (включно із розділом “Разновидности цикла *for*”), (англ. – р.105 ... 108, 166...171), [Хорстманн] – с. 93 (розділ 3.8.1.Область видимости блоков), с. 100...103 (розділ 3.8.4.Определенные циклы), (англ. – р.139, 140, 151...157).

**Завдання:** [Руденко] – вибрані завдання для самостійного виконання розділу 3.4.

**Урок 4.2.** Цикл *for* із внутрішніми операторами *if*.

**Відео:** Цикл *for* із внутрішніми операторами *if*.

**Завдання:**

1. [Руденко] – вибрані завдання для самостійного виконання розділу 3.4.
2. Визначити усі дільники довільного заданого натурального числа.
3. Визначити усі числа, які одночасно є дільниками кожного з двох довільних заданих натуральних чисел.
4. Визначити перші 10 чисел, що одночасно діляться на кожного з двох довільних заданих натуральних чисел.
5. Скільки існує натуральних чисел, менших 200, що не діляться ні на 5, ні на 7?
- 6\*. До числа 35 дописати ліворуч і праворуч по одній однаковій (інший варіант завдання - довільній) цифрі такій, щоб утворене число було кратним довільному заданому натуральному числу.
- 7\*. Знайти всі натуральні числа  $x$ , що задовольняють рівнянню

$$\underline{92abc = 874 \cdot x.}$$

Ліворуч стоїть п'ятицифрове число ( $a, b, c$  – цифри в межах  $0...9$ ).

8\*. Скільки існує пар  $(a, b)$  натуральних чисел, менших 200, таких, що сума  $a*a + b*b$  ділиться на 49?

9\*. Знайдіть всі натуральні числа, на які можна скоротити дріб  $(5x+6)/(8x+7)$  при різних цілих значеннях  $x$ .

10\*. Від тризначного числа відняли суму кубів його цифр. Який найбільший результат міг при цьому вийти?

11\*. Скільки існує натуральних чисел таких, які можна подати у вигляді суми двох тризначних чисел  $sba + abc$ , ( $a$  і  $c$  не рівні 0)?

### Урок 4.3. Оператори *while* та *do-while*.

**Відео:** Оператор циклу *while*.

**Література основна:** [Руденко] – с.49...53 (підрозділи “Цикли з передумовою”, “Цикли з післяумовою”).

**Література додаткова\*:** [Шилдт] – с.140...144 (розділи “Цикл *while*” та “Цикл *do-while*”), (англ. – р.160...166),  
[Хорстманн] – с. 97...100 (розділ 3.8.3.Неопределенные циклы), (англ. – р.145...151).

**Завдання:** [Руденко] – вибрані завдання для самостійного виконання розділу 3.4.

### Урок 4.4. Вкладені цикли.

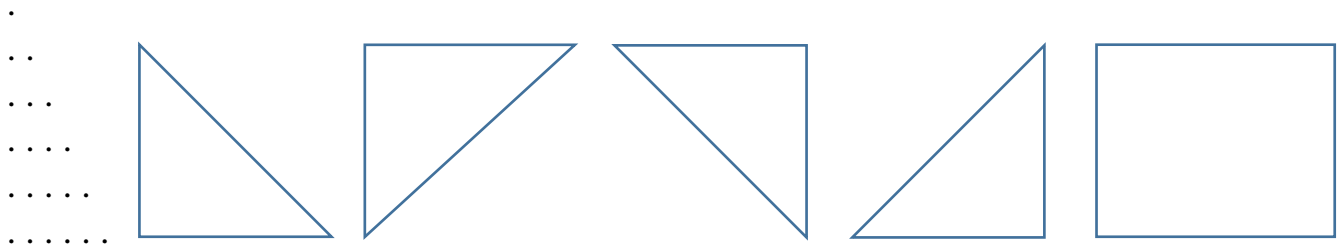
**Відео:** Вкладені цикли. Побудова крапкових трикутників.

**Література основна:** [Руденко] – с.57...58 (підрозділ “Вкладені цикли”).

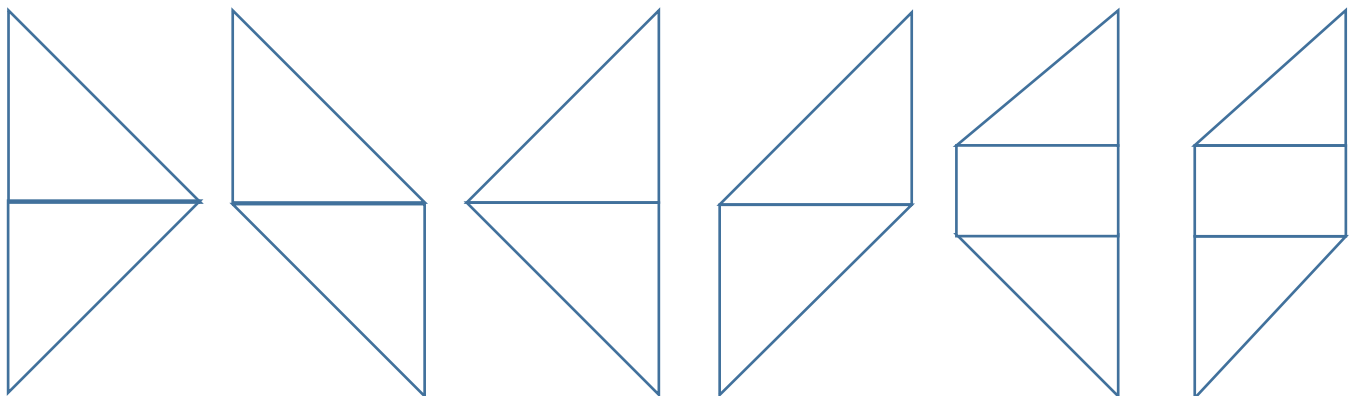
**Література додаткова:** [Шилдт] – с.154...155 (розділи “Вложенные циклы”), (англ. – р.178).

#### Проектні завдання:

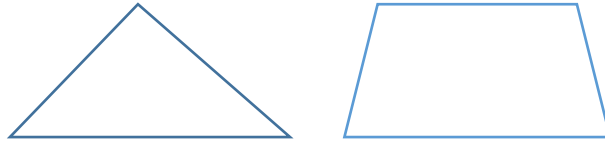
1. Реалізувати виведення примітивних крапкових “вертикальних” трикутників (Шилдт, ст.154...155) нижченаведених конфігурацій, а також прямокутника:



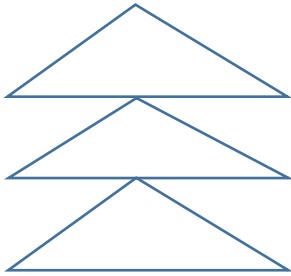
2. Із використанням алгоритмів виведення примітивних “вертикальних” трикутників та прямокутника, створених у завданні 1, реалізувати формування складених рівнобедрених “вертикальних” трикутників, паралелограмів та “вертикальних” трапецій наступних конфігурацій:



3\*.Реалізувати виведення примітивних крапкових “горизонтальних” трикутника та трапеції:



4\*. Із використанням алгоритму виведення примітивного “горизонтального” трикутника, створеного у завданні 3, реалізувати виведення крапкових трикутників, розміщених вертикально (“ялинку”):



Передбачити можливість побудови “ялинки” з трикутників різного розміру та з різними символами формування (“крапками”), зміну кількості трикутників для формування “ялинки”, а також можливість зсуву “ялинки” вправо на задану кількість кроків.

## ТЕМА 5

**Види масивів, способи їх опису. Одновимірні масиви. Поле *length*. Обробка масивів без розгалужень. Обробка масивів із розгалуженням. Обробка масивів із утворенням масивів-результатів відомої та невідомої довжин.**

(Для всіх програмних завдань теми 5 по опрацюванню масивів цілих чисел припускаємо, що елементи в масивах не повторюються).

**Урок 5.1.** Види масивів, способи їх опису. Одновимірні масиви. Поле *length*.

**Відео:** Поняття масиву. Виведення елементів масиву.

**Література основна:** [Руденко] – с.68...72 (розділ 4.1.Одновимірні масиви).

**Література додаткова\*:** [Шилдт] – с.99...102 (розділ “Одномерные массивы”), с.107 (розділ “Альтернативный синтаксис объявления массивов”), (англ. – р.112...115, 122...123), [Хорстманн] – с. 111...113 (розділи 3.10.1.Объявление массивов, 3.10.2. Доступ к элементам массива), (англ. – р.168...171).

**Завдання:**

- 1.[Руденко] – вибрані завдання для самостійного виконання розділу 4.1.
- 2.[Jaworsky] - р.72: 20,\*
- 3.[Heller] - Assesment Test: 11.\*

**Урок 5.2.** Обробка масивів без розгалужень.

**Відео:** Сума елементів масиву цілих чисел.

**Література основна:** [Руденко] – с.77 (підрозділ “Обчислення суми елементів масиву і середнього значення елементів масиву”).

### **Завдання:**

1. [Руденко] – вибрані завдання для самостійного виконання розділу 4.3.
2. З'єднати в одну текстову змінну всі елементи масиву String [ ].

### **Урок 5.3. Обробка масивів із розгалуженнями.**

**Відео:** Пошук максимуму в масиві цілих чисел.

**Література основна:** [Руденко] – с.77...78 (підрозділ “Визначення кількості елементів, які відповідають заданим умовам”), с.84...85 (підрозділ “Лінійний пошук даних”).

### **Завдання:**

1. [Руденко] – вибрані завдання для самостійного виконання розділів 4.3, 4.5.
2. Порахувати кількість та суму парних елементів масиву натуральних чисел.
3. Визначити максимальний з від’ємних та мінімальний з додатніх елементів масиву.

### **Урок 5.4. Обробка масивів із утворенням масивів-результатів відомої довжини.**

**Відео:** Попарне додавання елементів масивів цілих чисел.

**Література основна:** [Руденко] – с.78...80 (підрозділи “Формування нового масиву на основі значень елементів заданого масиву”, “Порівняння масивів”).

### **Завдання:**

1. [Руденко] – вибрані завдання для самостійного виконання розділу 4.3.
2. Збільшити на одиницю додатні елементи масиву та зменшити на одиницю від’ємні елементи масиву із утворенням нового масиву.
3. В масиві поміняти місцями максимум та мінімум із утворенням нового масиву.

### **Урок 5.5. Конкатенація масивів.\***

1. В масиві усунути максимум та мінімум із утворенням нового масиву.\*

2. Визначити середнє арифметичне елементів масиву та кількість елементів цього масиву, що є більшими за середнє арифметичне.\*

### **Урок 5.6. Поділ масиву.\***

1. Масив поділити на масив парних та непарних чисел.\*

2. Задано посортований за зростанням масив *numbers* цілих чисел та масив *levels* з простими числами. Необхідно поділити масив *numbers* на підмасиви цілих чисел, що знаходяться в межах між елементами масиву *levels* (менший рівень – включно, більший рівень - виключно). Обрати найбільш швидкий алгоритм.\*

## ЛІТЕРАТУРА

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. Шилдт Герберт. Java. Полное руководство. 10-е изд.: Пер. с англ. – СПб.: ООО “Альфа-книга”, 2018. -1488 с.
3. Хорстманн Кей С. Java. Библиотека профессионала. 11-е изд.: – СПб.: Диалектика, 2019. Т.1. Основы. - 864 с.
4. Хорстманн Кей С. Java. Библиотека профессионала. 11-е изд.: – СПб.: Диалектика, 2020. Т.2. Расширенные средства программирования. - 864 с.
5. Эккель Брюс. Философия Java. – СПб.: Питер, 2009. -640 с.
6. Вязовик Н.А. Программирование на Java. - М.: ИНТУИТ, 2016.
7. Іщеряков С.М. Комп'ютерна схемотехніка.–Івано-Франківськ: Полум'я,2011.– 114 с.
8. S.G.Ganesh, Tushar Sharma. Java SE7 Programmer Exams 1Z0-804 and 1Z0-805. A Comprehensive OCPJP 7 Certification Guide.– APRESS,New York,2012.-644 p.
9. Philip Heller, Simon Roberts. Complete Java 2 Certification Study Guide. 5-th edition. – SYBEX, Inc., Alameda, 2005.- 560 p.
- 10.Raposa, Richard F. SCJP Sun Certified Programmer for Java Platform, SE6, Study Guide. – Wiley Publishing, Inc., Indianapolis, 2009.- 583 p.
- 11.Khalid A.Mughal, Rolf W.Rasmussen. A Programmer's Guide to Java SE8 Oracle Certified Associated (OCA). – Addison-Wesley Publishing, 2017.
- 12.Kathy Sierra, Bert Bates. SCJP Sun Certified Programmer for Java 5 Study Guide. – McGraw Hill Osborne, New York, 2006. -924 p.
- 13.Kathy Sierra, Bert Bates. SCJP Sun Certified Programmer for Java 6 Study Guide. – McGraw Hill Osborne, New York, 2008. -852 p.
- 14.Kathy Sierra, Bert Bates. OCA/OCP Java SE7 Programmer I & II Study Guide (Exams 1Z0-803 & 1Z0-804) (Oracle Press).– McGraw Hill Osborne, New York, 2015. -1088 p.
- 15.Bert Bates, Kathy Sierra. OCP Java SE6 Programmer Practice Exam.– McGraw Hill Osborne, New York, 2011. -422 p.
- 16.Jamie Jaworsky. Java 2 Certification Training Guide. – New Riders Publishing, Indianapolis, 1999.- 399 p.
- 17.Mala Gupta. OCA Java SE7 Programmer I Certification Guide. Prepare for the 1Z0-803Exam. – Manning, Shelter Island, 2013.- 528 p.
- 18.Paul Sanghera. SCJP Exam for J2SE5. A Concise and Comprehensive Study Guide for the Sun Certified Java Programmer Exam. – APRESS, New York,2006.- 423 p.
- 19.Jeanne Boyarsky, Scott Selikoff. OCP Oracle Certified Professional Java SE8 Programmer II. Study Guide. Exam 1Z0-809- John Wiley & Sons, Inc., Indianapolis, 2016.- 669 p.
- 20.Richard M.Reese. Oracle Certified Associate Java SE7 Programmer Study Guide. – Packt Publishing, Birmingham, 2012.- 313 p.