

Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Методичні вказівки

до виконання самостійних робіт
з дисципліни «Об'єктно-орієнтоване програмування»
для студентів спеціальності
123 «Комп'ютерна інженерія»
усіх форм навчання

Кривий Ріг

2021

Укладачі: Музика І. О., канд. техн. наук, доцент

Кузнєцов Д. І., канд. техн. наук, доцент

Рецензент: Тиханський М. П., канд. техн. наук, доцент

Дані методичні вказівки містять завдання самостійної роботи з дисципліни «Об'єктно-орієнтоване програмування» для студентів спеціальності 123 «Комп'ютерна інженерія». Представлено 300 завдань, які охоплюють базові теми об'єктно-орієнтованого програмування: елементарні та керуючі операції, масиви, рядки, роботу з файлами та візуальне програмування мовою С#.

Розглянуто
на засіданні кафедри
комп'ютерних систем та мереж

Протокол № 1
від 27.08.2021 р.

Схвалено
на вченій раді факультету
інформаційних технологій

Протокол № 1
від 30.08.2021 р.

ВСТУП

Об'єктно-орієнтоване програмування (ООП) – одна з парадигм програмування, яка розглядає програму як множину «об'єктів», що взаємодіють між собою. В ній використано декілька технологій, зокрема успадкування, модульність, поліморфізм та інкапсуляцію. Незважаючи на те, що ця парадигма з'явилась ще в 1960-их роках, вона не мала широкого застосування до 1990-их. Сьогодні багато мов програмування (зокрема, Java, C#, C++, Python, PHP, Ruby, Objective-C, Action Script 3 та ін.) підтримують ООП.

Дисципліна «Об'єктно-орієнтоване програмування» – одна із фундаментальних дисциплін професійного спрямування у загальній системі підготовки студентів спеціальності 123 «Комп'ютерна інженерія». Передумовою успішного вивчення даної дисципліни є володіння навичками програмування та алгоритмізації, які отримані під час вивчення дисциплін першого року навчання: «Основи інформаційних технологій», «Вища математика» та «Програмування».

Метою даної дисципліни є оволодіння мовою програмування C#, а також методами об'єктно-орієнтованого проектування та розробки складних програм. Як результат навчання за курсом «Об'єктно-орієнтоване програмування» студенти повинні: знати основи побудови об'єктів, способи обробки виключних ситуацій, читання та запису файлів, вміти будувати додатки типу Console, Windows Forms на базі .NET Framework з використанням середовища Microsoft Visual Studio.

Запропонований матеріал містить варіанти індивідуальних завдань з основних розділів дисципліни «Об'єктно-орієнтоване програмування». Студент виконує дані завдання протягом навчального семестру та отримує бали відповідно до розподілу, передбаченого робочою програмою навчальної дисципліни. Студент може звітувати про свою самостійну роботу, включаючи вирішення даних завдань до звітів з лабораторних робіт, або на модульних контролях.

ВАРІАНТИ ІНДИВІДУАЛЬНИХ ЗАВДАНЬ

№ варіанту	Задачі для розв'язання за відповідними темами						
	Умовні оператори, цикли	Одновимірні масиви	Двовимірні масиви	Символьні рядки	Класи, принципи ООП	Робота з файлами	Візуальне програмування
1	10, 17, 25	79, 89, 91	106, 115, 116	133, 141, 167	185, 201, 215	223, 239, 256	290, 292, 297
2	20, 26, 30	82, 85, 96	110, 112, 127	140, 177, 179	184, 195, 212	238, 255, 257	272, 285, 290
3	27, 41, 54	83, 100, 101	123, 125, 130	145, 150, 171	214, 215, 218	223, 236, 240	291, 292, 294
4	4, 9, 60	78, 83, 97	106, 112, 123	144, 156, 173	200, 210, 219	229, 253, 255	271, 289, 290
5	4, 34, 45	76, 78, 80	119, 127, 131	135, 154, 166	206, 209, 214	230, 244, 263	269, 272, 298
6	12, 47, 49	84, 93, 100	110, 118, 122	138, 142, 177	186, 215, 220	238, 252, 255	272, 294, 298
7	1, 37, 66	81, 85, 94	103, 105, 124	134, 162, 167	197, 210, 216	224, 246, 254	274, 275, 279
8	22, 39, 57	78, 92, 102	113, 116, 130	141, 142, 157	191, 194, 216	223, 244, 257	280, 289, 290
9	30, 41, 46	84, 93, 94	115, 117, 128	142, 161, 183	197, 218, 221	239, 242, 244	286, 291, 300
10	16, 27, 29	78, 91, 95	105, 115, 122	136, 153, 162	196, 205, 220	224, 261, 265	270, 290, 298
11	10, 40, 71	83, 98, 102	110, 113, 123	136, 138, 177	187, 206, 216	230, 237, 255	275, 283, 297
12	25, 50, 72	76, 77, 94	104, 109, 127	137, 150, 169	190, 208, 221	227, 235, 250	278, 281, 300
13	15, 34, 71	85, 88, 97	107, 117, 120	150, 154, 176	195, 210, 214	228, 246, 259	275, 278, 300
14	8, 59, 62	81, 84, 92	105, 116, 130	135, 142, 165	201, 207, 211	231, 235, 260	280, 282, 283
15	1, 5, 48	79, 80, 102	104, 106, 127	139, 164, 167	202, 218, 219	228, 241, 261	269, 283, 300

№ варіанту	Задачі для розв'язання за відповідними темами						
	Умовні оператори, цикли	Одновимірні масиви	Двовимірні масиви	Символьні рядки	Класи, принципи ООП	Робота з файлами	Візуальне програмування
16	57, 67, 72	79, 91, 92	103, 106, 119	156, 169, 175	194, 200, 210	223, 246, 251	270, 274, 285
17	4, 14, 33	78, 84, 85	109, 116, 122	137, 140, 145	199, 209, 216	226, 245, 247	270, 276, 298
18	9, 52, 56	77, 80, 102	107, 109, 111	143, 150, 154	206, 207, 208	239, 251, 256	276, 287, 300
19	4, 38, 62	87, 89, 91	125, 131, 132	133, 158, 167	192, 194, 208	234, 241, 256	270, 271, 278
20	60, 62, 72	78, 92, 94	118, 128, 132	141, 148, 170	205, 210, 217	236, 240, 264	273, 282, 286
21	12, 28, 45	80, 82, 92	105, 110, 127	142, 150, 171	197, 201, 221	239, 244, 263	272, 290, 293
22	14, 40, 70	81, 94, 98	105, 116, 127	139, 152, 179	187, 194, 196	227, 238, 268	275, 285, 297
23	8, 21, 59	79, 83, 86	103, 112, 129	162, 168, 169	188, 203, 220	223, 262, 264	270, 288, 295
24	34, 52, 53	80, 91, 100	105, 113, 126	150, 156, 172	184, 209, 215	229, 233, 258	272, 288, 293
25	7, 15, 29	76, 96, 102	108, 114, 121	158, 180, 182	187, 198, 208	234, 246, 264	279, 291, 297
26	9, 17, 39	78, 88, 89	109, 119, 122	135, 160, 173	199, 211, 220	230, 233, 252	274, 280, 290
27	34, 59, 64	77, 80, 93	111, 117, 129	156, 159, 176	197, 202, 209	226, 241, 256	281, 285, 299
28	3, 38, 69	81, 82, 97	103, 109, 122	158, 162, 174	195, 210, 219	238, 246, 266	269, 275, 289
29	6, 62, 64	83, 88, 99	103, 105, 119	166, 167, 181	193, 202, 212	248, 262, 266	295, 297, 300
30	62, 68, 70	85, 89, 96	105, 116, 129	133, 153, 168	184, 220, 222	250, 262, 268	276, 297, 298

1. УМОВНІ ОПЕРАТОРИ, ЦИКЛИ

Задача 1. Написати програму обчислення суми і добутку двох чисел.

Задача 2. Написати програму знаходження гіпотенузи і площі прямокутного трикутника за двома катетами.

Задача 3. Написати програму обчислення периметра і площі прямокутного трикутника за двома катетами.

Задача 4. Програма запитує у користувача a , b , c . Обчислити значення виразу $a^2 - (b-c)^2 + \ln(b^2+1)$.

Задача 5. Програма запитує три числа a , b , c . Вивести на екран значення виразів $a^2 + b + 1$ (червоним кольором), $a + b + 2\sqrt{c}$ (зеленим) і $a + \sqrt{bc} + 5c$ (жовтим).

Задача 6. Дано дійсне число, після його округлення перевірити: Якщо воно кратне 4, то друкувати «потрібне число»; Якщо воно кратне 3 і 5, то також друкувати «потрібне число», якщо число парне, то друкувати «стороннє число», якщо парне число не кратне 6, то вивести на екран повідомлення « парне число не кратне 6».

Задача 7. Програма вводить два числа a і b , якщо $a > b$, то Програма вводить третє число c . Якщо $a = b$, то Програма друкує слово «Кінець». Якщо $a < b$, то Програма вводить третє число c , якщо $c < 8$, то Програма знаходить значення виразу $\ln(a^2 + b^2) + \exp(c)$, виводить його і слова «Новий год» на екран.

Задача 8. Програма вводить ціле число n , якщо воно >0 і парне, Програма вводить друге число. Якщо обидва числа парні, то Програма виводить на екран суму 2-х чисел, во всіх чисел останніх випадках Програма виводить на екран повідомлення «Кінець роботи».

Задача 9. Програма вводить три числа a , b , c . Якщо $a-b > c$, то вивести слово «Програмування», в останніх випадках вивести на екран «Кінець роботи» зеленим кольором.

Задача 10. Програма вводить два числа a і b , якщо $a > b$, то Програма вводить c і виводить на екран значення виразу $a^7 + \frac{ab}{19} + c^{\frac{1}{5}}$, інакше – виводить різницю $a - b$ і словосполучення «Кінець роботи».

Задача 11. Програма вводить ціле число a , якщо $a > 5$, то вивести на екран вираз $a + b$, інакше вивести на екран слово «комп'ютер».

Задача 12. Дано n дійсних чисел. Знайти кількість тих із них, котрі після усікання кратні 5 або 6, і суму чисел, котрі більше 4 і їх номер кратний 3.

Задача 13. Дано N дійсних чисел. Після усікання визначити кількість парних, кількість кратних 3, кількість чисел, котрі діляться на 5 з остачею 2.

Задача 14. Дано n дійсних чисел. Знайти кількість тих із них, котрі після усікання стали більше 15, парні і кратні 5.

Задача 15. Дано n дійсних чисел. Знайти кількість тих із них, котрі після округлення стали парними, і суму тих, котрі більше 54.

Задача 16. Дано N дійсних чисел. Визначити, скільки чисел попадають в інтервал від 3 до 8, скільки – після усікання кратні 7, скільки парних, скільки чисел, котрі після округлення не стали кратними 11 і 2 одночасно.

Задача 17. Програма запитує кількість цілих чисел, потім вводить цілі числа згідно з вказаною кількістю, знаходить середнє арифметичне і суму від'ємних чисел.

Задача 18. Програма запитує кількість чисел, потім вводить цілі числа згідно з вказаною кількістю. Програма знаходить кількість чисел, котрі по модулю > 5 , кількість чисел, кратних 4 і 5 одночасно, і суму чисел, котрі діляться на 6 з остатком 5.

Задача 19. Програма запитує кількість чисел, потім вводить цілі числа згідно з вказаною кількістю. Визначає кількість парних, суму кратну 3-м і кількість чисел, модуль яких < 3 .

Задача 20. Програма запитує, скільки чисел хоче скласти користувач. Програма просить ввести ці числа, а потім виводить отриману суму.

Задача 21. Програма запитує кількість чисел, потім вводить цілі числа згідно з вказаною кількістю. Програма знаходить суму від'ємних і суму непарних чисел. Програма виводить ту суму, яка за модулем менша.

Задача 22. Дано N дійсних чисел. Після округлення визначити кількість парних, кількість кратних 7, кількість чисел, котрі діляться на 8 з остатком 7.

Задача 23. Програма запитує у користувача кількість чисел, вводить цілі числа згідно із заданою кількістю. Знайти кількість непарних чисел і вивести зеленим кольором значення виразу

$$\sin \sqrt{kol^2 + \frac{1}{5}}.$$

Задача 24. Програма запитує у користувача кількість чисел, вводить дійсні числа. Знайти суму тільки тих чисел, котрі < 0 , і вивести значення жовтим кольором.

Задача 25. Написати програму, яка із N довільних дійсних чисел знаходить максимум, середнє арифметичне, а після округлення чисел визначає кількість чисел, кратних 7.

Задача 26. Дано N дійсних чисел. Знайти мінімум серед додатних чисел і максимум серед від'ємних, надрукувати найбільше по абсолютній величині.

Задача 27. Задана довільна послідовність символів. Ознака кінця – «/». Розфарбувати сполучення символів «nn» бузковим кольором.

Задача 28. Програма виводить довільну послідовність символів до тих пір, поки це цифри. Програма визначає, скільки раз зустрілася цифра «2», друкує тричі всі цифри 7 з нового рядка, а останні цифри на екран не виводить.

Задача 29. Написати програму, яка обчислює вираз $x^{50} - y^{20} + z^n$. Програма запитує у користувача x , y , z і n .

Задача 30. Написати програму, яка обчислює вираз $\sqrt{x^{18} + 1} - \sqrt{y^{18} + 16}$. Програма запитує у користувача значення x і y .

Задача 31. Програма виводить довільні цілі числа до тих пір, поки їх сума не перевищить 20. Визначити кількість чисел, при яких виконується ця умова. Провести виведення проміжних результатів.

Задача 32. Дано 10 випадкових чисел в діапазоні від -146 до 134 . Обчислити різницю між максимальним і мінімальним елементами.

Задача 33. Задано координати вершин двох трикутників. Визначити, який із них має більшу площу.

Задача 34. Задана довільна послідовність символів. Ознака кінця – «/». Визначити, скільки раз в ній зустрічається сполучення «що».

Задача 35. Дано N дійсних чисел. Знайти максимум серед додатних і мінімум серед від'ємних, вивести на екран найбільше за абсолютною величиною.

Задача 36. Задано три числа x, y, z . Написати програму для знаходження u :
$$u = \frac{\max^2(x, y, z) - 2^x \min(x, y, z)}{\sin 2 + \max(x, y, z) / \min(x, y, z)}$$

Задача 37. Задано три числа x, y, z . Написати програму для знаходження u :
$$u = \frac{\max^2(y, z) - 2^x \min(x, y)}{\sin 2 + \max(x, y, z) * \min(y, z)}$$

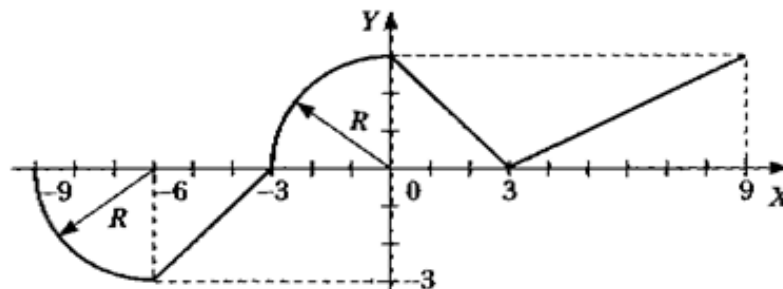
Задача 38. Дано натуральне число n . Обчислити: $(1-1/2^2)*(1-1/3^2)*(1-1/4^2)*\dots*(1-1/n^2)$.

Задача 39. Задано числа x, y . Знайти:
$$z = \begin{cases} \max(x, y), & \text{якщо } \cdot y > 0 \\ \min(x, y), & \text{якщо } \cdot y < 0 \end{cases}$$

Задача 40. Задано числа x, y, z . Переприсвоїти їх значення так, щоб $x^2 > y^2 > z^2$.

Задача 41. Знайти всі натуральні числа, які не перевищують задане число N , десятковий запис яких є строго зростаюча або строго спадаюча послідовність цифр.

Задача 42. Написати програму, яка за введеним значенням аргументу обчислює значення функції, заданої у вигляді графіка. Параметр R вводиться з клавіатури.



Задача 43. Два натуральних числа називаються дружніми, якщо кожне із них рівне сумі всіх дільників іншого, крім самого цього числа. Знайти всі пари дружніх чисел в діапазоні від 200 до 300.

Задача 44. Отримати всі шестизначні щасливі номери (тобто такі номери, у яких сума перших трьох цифр рівна сумі трьох останніх).

Задача 45. Задано дійсні додатні числа a, b, c, d . З'ясувати, чи можна побудувати чотирикутник з такими довжинами сторін.

Задача 46. Знайти суму всіх дільників числа.

Задача 47. Знайти суму площ трикутників ABC і MNP , заданих координатами вершин. Дані для вводу: $A(0; 1), B(3; 1), C(4, 2), M(6, 7), N(4, 3), P(3; 8)$.

Задача 48. Знайти суму периметрів трикутників ABC і MNP , заданих координатами вершин. Дані для вводу: $A(0; 1), B(2; 4), C(2, 8), M(6, 7), N(0, 3), P(3; 8)$.

Задача 49. Обчислити $1! + 2! + 3! + \dots + N!$.

Задача 50. Визначити довжину кола C і площу круга S , відстань L центру кола від початку координат O . Координати центру кола рівні X і Y , радіус R .

Задача 51. Скласти підпрограму для визначення значень $n!, m!, (n*m)!$

Задача 52. З клавіатури вводяться числа, до тих пір, поки не буде введено перше від'ємне число. Визначити, скільки чисел із вхідного потоку, дорівнює сумі кубів своїх цифр.

Задача 53. Написати функцію логічного типу, яка перевіряє, чи є різними всі цифри, що входять в натуральну запис, числа N .

Задача 54. Скласти програму підрахунку кількості всіх натуральних чисел, менших M і які діляться на кожну зі своїх цифр.

Задача 55. Скласти програму знаходження найменшого натурального N -значного числа $X \geq 10$, рівного потроєному добутку своїх цифр.

Задача 56. Дано натуральне число. Визначити, скільки парних цифр використовується в запису цього числа.

Задача 57. Скласти програму підрахунку кількості всіх натуральних чисел, менших M , квадрат суми цифр яких дорівнює X .

Задача 58. Від користувачу необхідно ввести два числа. Ці значення повинні бути присвоєні двом змінним, причому значення першої повинно бути менше другої. Звичайно, можна попередити про це користувача, але можливо йому зручніше буде і не знати про вимоги програми. У кодї програми можна реалізувати процедуру, яка виробляє обмін значень двох змінних, і викликати її після кожного чергового введення користувача.

Задача 59. Визначити члени послідовності Фібоначчі.

Задача 60. Записати власну функцію обчислення модуля числа n `Modul(n)`.

Задача 61. Написати процедуру, яка під час натискання клавіші пробіл видає інформацію про розробника програми.

Задача 62. Дано дійсні числа a, b, c, d . Визначити значення: $y = (p(a) + p(b) + p(c) + p(d))/4$, де $p(x) = 4x^4 + 3x^3 + 2x^2 + x + 0.5$.

Задача 63. Дано дві пари дійсних чисел $(x_1; y_1)$ та $(x_2; y_2)$, які є координатами відповідно лівого верхнього і правого нижнього кутів прямокутника, сторони якого паралельні осям координат. Створити програму, що визначатиме довжини сторін цього прямокутника.

Задача 64. Створити підпрограму, яка б за заданою стороною квадрата a визначала периметр квадрата і його діагональ.

Задача 65. Створити підпрограму, яка б за даними дійсними координатами двох точок $(x_1; y_1)$ та $(x_2; y_2)$ визначала довжину відповідного відрізка та довжину його проекції на вісь OX . У програмі організувати виклик цієї підпрограми.

Задача 66. Створити підпрограму, що одночасно обчислює найменше і найбільше значення серед двох дійсних чисел, та використати її для визначення, у скільки разів найменше значення з трьох заданих дійсних чисел перевищує найменше з них

Задача 67.* Дано ціле число, записане в системі числення з основою 7. Перевести його в систему числення з основою 5.

Задача 68.* Знайти всі прості числа, які не перевищують задане N , двійковий запис яких представляє собою симетричну послідовність нулів і одиниць.

Задача 69.* Серед простих чисел, які не перевищують задане N , знайти таке, в двійковому представленні якого максимальне число одиниць.

Задача 70.* Для натуральних чисел A, B операцію \oplus визначимо так: $A \oplus B = A - B + (A \bmod B)$. Знайти всі такі пари A, B , які не перевищують задане число N , для яких $A \oplus B = B \oplus A$.

Задача 71.* Дано ціле число k від 1 до 180. Визначити, яка цифра знаходиться в k -ій позиції послідовності: 101112...9899, в якій виписані підряд все двозначні числа.

Задача 72.* Дано ціле число k . Визначити, яка цифра «0» або «1» знаходиться в k -ій позиції послідовності: 11010010001..., в якій виписані підряд степені числа 10.

Задача 73.* Задана множина точок у тривимірному просторі. Знайти мінімум радіусів шарів з центрами в цих точках, які містять рівно n точок цієї множини.

Задача 74.* Указати те число заданої множини цілих чисел, в двійковому представленні якого більше всього одиниць.

Задача 75.* Багатокутник (не обов'язково випуклий) задано на площині перерахуванням координат вершин у порядку обходу його сторін. Визначити площу багатокутника.

2. ОДНОВИМІРНІ МАСИВИ

Задача 76. Програма заповнює одновимірний масив із 10 цілих чисел числами, які зчитуються з клавіатури. Програма визначає суму додатних чисел і середнє арифметичне всіх чисел масиву.

Задача 77. Програма заповнює одновимірний масив із 20 дійсних чисел з клавіатури. Програма знаходить суму від'ємних чисел, відсікає дробову частку чисел і визначає, скільки серед них парних, скільки чисел за модулем більше 5.

Задача 78. Програма заповнює масив із 20 цілих чисел випадковими числами в діапазоні від -3 до 18. Програма знаходить максимальне число.

Задача 79. Знайти три найбільших елемента одновимірного масиву із 30-ти елементів.

Задача 80. Програма заповнює одновимірний масив із 20-ти елементів випадковими числами в діапазоні від -5 до 10. Програма визначає кількість таких елементів в масиві, для яких попередній елемент менше, а наступний більше даного елемента.

Задача 81. Дано одновимірний масив із 25-ти символів. Вивести елементи масиву на екран так, що всі цифри були виведені зеленим кольором, а останні символи – червоним.

Задача 82. Програма вводить два цілих числа $n_1 < n_2 <$ розміру масиву. Відсортувати бульбашкою масив між елементами з номерами n_1 і n_2 .

Задача 83. Обчислити кількість точок з цілими координатами, які попадають в коло радіуса $R > 0$ з центром у початку координат. Координати точок знаходяться в масиві.

Задача 84. В одновимірному масиві, який складається із 20-ти символів, замінити цифри на «*», великі букви латиниці – на «!», а на місце прописний латинських букв поставити «~». Останні символи оставить без змін. Вивести на екран елементи масиву до і після проведених змін.

Задача 85. Одновимірний масив із 10-ти цілих чисел заповнити з клавіатури, визначити суму тих чисел, котрі більші 5 і є парними.

Задача 86. Заповнити одновимірний масив 15-ма символами. У масиві із символів всі цифри замінити на «*». Вивести елементи зміненого масиву.

Задача 87. Дано одновимірний масив із дійсних чисел. Ввести число c і знайти значення виразу $c + \frac{c}{a_1} + \frac{c}{a_2} + \dots + \frac{c}{a_n}$, де a – елементи масиву.

Задача 88. Програма заповнює масив із 12-ти цілих чисел випадковими числами в діапазоні від -3 до 38 . Програма знаходить максимальний елемент масиву і виводить його на екран.

Задача 89. Дано два одновимірних цілочисельних масиви довжиною 10 і 15 елементів. Роздрукувати їх в 2 і 3 колонки відповідно. Надрукувати елементи першого масиву, повторювані у другому.

Задача 90. Дано масив із 30-ти цілих елементів. Надрукувати його у вигляді таблиці, де кількість рядків і стовпчиків замовляє користувач.

Задача 91. Знайти відхилення кожного елемента $a[i]$ цілочисельного масиву a від середнього арифметичного значення цього масиву.

Задача 92. Надрукувати масив із 10-ти дійсних елементів у зворотному порядку.

Задача 93. Надрукувати масив із 30-ти дійсних елементів в чотири колонки.

Задача 94. Заповнити випадковими числами одновимірний масив із 20-ти елементів. Зафарбувати масив при виведенні на екран зеленим кольором за виключенням елементів з номерами від n_1 до n_2 , які зафарбувати жовтим кольором.

Задача 95. У одновимірному масиві, що складається з n дійсних елементів, обчислити: суму від'ємних елементів масиву; добуток елементів масиву, розташованих між максимальним і мінімальним елементами.

Задача 96. Упорядкувати елементи масиву за збільшенням та за зменшенням.

Задача 97. У одновимірному масиві з n дійсних елементів обчислити: суму додатних елементів масиву; добуток елементів масиву, розташованих між першим і останнім від'ємними елементами.

Задача 98. Стиснути масив, видаливши з нього всі елементи, модуль яких не перевищує одиницю. Елементи, що знаходяться в кінці масиву, заповнити нулями.

Задача 99. Перетворити масив так, щоб спочатку розташовувалися всі елементи, рівні нулю, а потім – всі останні.

Задача 100. Перетворити масив так, щоб спочатку розташовувалися всі від'ємні, а потім – всі додатні елементи (елементи, рівні нулю, вважати додатними).

Задача 101. Перетворити масив так, щоб спочатку розташовувалися всі елементи, що відрізняються від максимального не більше, ніж на 20%, а потім – всі останні.

Задача 102. Задано масив. Упорядкувати за збільшенням окремо елементи, що стоять на парних місцях, і елементи, що стоять на непарних місцях.

3. ДВОВИМІРНІ МАТРИЦІ

Задача 103. Задано матриці $A[m \times n]$ і $B[m \times n]$. Програма виводить матрицю $C = A + B$, зафарбовуючи нулі в зелений колір.

Задача 104. Задано матриці $A[6 \times 6]$ і $B[6 \times 6]$. Програма визначає, скільки додатних елементів знаходиться у матриці $C = A + B$.

Задача 105. Задано дві матриці $A[m \times n]$ і $B[m_1 \times n_1]$. Програма знаходить і роздруковує суму елементів 5-го стовпчика матриці A і суму елементів останнього рядка матриці B .

Задача 106. Задано дві матриці $A[m \times n]$ і $B[m_1 \times n_1]$. Програма знаходить і виводить на екран суму елементів останнього стовпчика матриці A і суму елементів 2-го рядка матриці B .

Задача 107. Задана матриця $A[m \times n]$, заповнена випадковими числами. Програма знаходить максимальний елемент i -ого рядка і виводить його на екран.

Задача 108. Програма вводить цілу матрицю $A[m \times n]$ і виводить транспоновану до неї матрицю A^T , всі елементи якої зменшені удвічі.

Задача 109. Задано цілочисельні матриці $A[m \times n]$ і $B[m \times n]$. Програма знаходить кількість нулів справа від головної діагоналі в матриці $C = A + B$, виводить на екран матрицю C і кількість нулів.

Задача 110. Прямокутна матриця $X[4 \times 4]$ заповнена цілими числами з клавіатури. Вивести на екран матрицю, зафарбувавши максимальний елемент зеленим кольором.

Задача 111. Програма знаходить середнє арифметичне дійсної матриці $A[m \times n]$.

Задача 112. Заповнити дійсну матриця $A[m \times n]$ з клавіатури. Вивести матрицю на екран у вигляді таблиці, так щоб елементи матриці, котрі більші середнього арифметичного матриці були зафарбовані зеленим кольором, а менші – червоним.

Задача 113. Задана матриця $A[m \times n]$, заповнена випадковими числами. Програма знаходить максимальний елемент 2-го стовпчика.

Задача 114. Програма заповнює двовимірний масив $A[8 \times 8]$ випадковими числами в діапазоні від -12 до 23 і виводить його на екран

у вигляді таблиці, так щоб елементи другого рядка таблиці були зафарбовані зеленим кольором.

Задача 115. Задано два одновимірних масиви $A[1 \times 6]$ і $B[1 \times 5]$ і матриця $C[4 \times 5]$. Потрібно скласти програму друку всіх елементів матриці D_{ij} , котрі менше мінімального елемента матриці C . Елементи матриці D_{ij} розраховуються за формулою $D_{ij} = A_i * B_j$; $i = 1, 2, \dots, 6$; $j = 1, 2, \dots, 5$.

Задача 116. Дано двовимірний масив $A[8 \times 7]$, заповнений випадковими числами. Замінити всі елементи перших трьох стовпчиків на їх квадрати. Вставити між середніми рядками перший рядок. Видалити всі стовпчики, в яких перший елемент більше останнього.

Задача 117. Написати програму, яка заповнює матрицю $A[m \times n]$ цілими числами. Програма запитує цілі числа k і q , створює нову матрицю $B[(m-1) \times (n-1)]$ шляхом видалення k -го рядка і q -го стовпчика із матриці A і виводить у вигляді таблиці матриці A і B на екран.

Задача 118. Написати програму, яка вводить цілочисельну матрицю $A[7 \times 7]$ і виводить її на екран у вигляді таблиці, знаходить суму елементів головної діагоналі і кількість парних елементів матриці, розміщених зліва від головної діагоналі матриці.

Задача 119. Вивести на екран матрицю C , отриману множенням матриці $A[10 \times 11]$ на матрицю $B[11 \times 4]$.

Задача 120. Програма запитує розмірність матриці, але не більше 10×10 , при цьому перевіряє правильність введення числа рядків і числа стовпчиків. За бажанням користувача програма заповнює матрицю випадковими числами або вводить матрицю з клавіатури, перевіряючи правильність введення, запитує номер рядка матриці і виводить на екран максимум цього рядка.

Задача 121. Програма формує масив із сум елементів матриці $A[5 \times 8]$, які стоять в одному рядку. Матриця $A[5 \times 8]$ заповнена випадковими числами.

Задача 122. Програма формує одновимірний масив із сум елементів дійсного двовимірного масиву, які стоять в одному рядку і виводить його на екран так, що елементи масиву розміщуються поряд із рядком масиву.

Задача 123. Програма заповнює цілочисельну матрицю $A[m \times n]$ випадковими числами в діапазоні від -6 до 10 і виводить на екран у вигляді таблиці. Програма визначає номер рядка, в якому знаходиться мінімальний елемент матриці A . Якщо мінімумів декілька, то програма виводить на екран список номерів рядків, в яких знаходяться мінімуми.

Задача 124. Розглядаючи елементи масиву M як координати точок на площині, знайти найбільшу і найменшу відстань між цими точками.

Задача 125. Задано натуральне n і елементи квадратної дійсної матриці $A[5 \times 5]$. Обчислити n -ту степінь цієї матриці ($A^1 = A$, $A^2 = A * A$, $A^3 = A^2 * A$ і т. д.).

Задача 126. Задана дійсна матриця $A[5 \times 6]$. Впорядкувати її рядки за не спаданням суми їх елементів.

Задача 127. Характеристикою рядка цілочисельної матриці назвемо суму її додатних парних елементів. Переставляючи рядки заданої матриці, розташувати їх відповідно до зростання характеристик.

Задача 128. Задана цілочисельна квадратна матриця. Визначити: суму елементів в тих стовпцях, які не містять від'ємних елементів; мінімум серед сум модулів елементів діагоналей, паралельних побічній діагоналі матриці.

Задача 129. Для заданої матриці розміром 8×8 Знайти такі k , при яких k -й рядок збігається з k -м стовпцем.

Задача 130. Дано матрицю. Знайти суму елементів в тих рядках, які містять хоч би один від'ємний елемент.

Задача 131. Характеристикою стовпця цілочисельної матриці назвемо суму модулів його від'ємних непарних елементів. Переставляючи стовпці заданої матриці, розташувати їх відповідно до зростання характеристик.

Задача 132. Ущільнити задану матрицю, видаляючи з неї рядки і стовпці, заповнені нулями.

4. СИМВОЛЬНІ РЯДКИ

Задача 133. У заданому тексті замінити слово *A* на слово *B* (довжини слів не співпадають).

Задача 134. У заданому тексті визначити кількість слів.

Задача 135. Маємо відомість, яка складається із 10 прізвищ. Знайти і надрукувати порядкові номери прізвищ, які задаються.

Задача 136. У слові «лірика» після кожного складу вставити склад «ма».

Задача 137. У тексті забрати зайві пропуски між словами, залишивши по одному.

Задача 138. У тексті вставити між словами замість одного два пропуски.

Задача 139. У заданому слові переставити літери в алфавітному порядку.

Задача 140. У текстовій змінній «71D523CE8» поставити числа у порядку зменшення.

Задача 141. Розділити заданий текст на рядки. У якості розділення рядків використаний символ %.

Задача 142. У виразі «місячний шлях» переставити місцями слова.

Задача 143. Дано слово «алгоритм». Вивести його на друк у зворотному порядку.

Задача 144. У слові «електроніка» переставити літери у порядку зворотному алфавіту.

Задача 145. Надрукувати найдовше слово із заданого тексту «Ціль статистики зосереджується в наданні фактів у найбільш стислій формі».

Задача 146. У текстовій змінній «2C35IA4» представити числа у порядку зростання.

Задача 147. Існує відомість, яка складається із 10 прізвищ. Вивести на друк цю відомість у алфавітному порядку.

Задача 148. В слові, що водить користувач, всі голосні літери видалити та на їх місце поставити пропуск. Вивести на дисплей початкове слово та результат перетворень.

Задача 149. У слові «Література» кожен голосну літеру виділити символом «.».

Задача 150. У заданому слові змінити місцями першу і останню літери. Вивести на екран дисплея початкове слово і результат.

Задача 151. У заданому тексті видалити частину тексту, яка взята в дужки разом з дужками.

Задача 152. Перевірити, чи присутній в заданому тексті баланс відкритих та закритих круглих дужок?

Задача 153. Шукати у реченні відкриті та закриті дужки. При знаходженні відкритої дужки збільшити лічильник на 1, при закритій – зменшити. Слідкувати за величиною лічильника.

Задача 154. У заданому реченні знайти пару слів, із яких одно є інверсією другого.

Задача 155. Для кожного із слів заданого речення вказати, скільки раз воно зустрічається в реченні.

Задача 156. Із заданого тексту вибрати і надрукувати ті символи, котрі зустрічаються в ньому рівно один раз (у тому порядку, як вони зустрічаються в тексті).

Задача 157. У реченні всі слова починаються з різних букв. Надрукувати (якщо можливо) слова речення у такому порядку, щоб остання буква кожного слова співпадала з першою буквою наступного слова.

Задача 158. Знайти множину всіх слів, які зустрічаються у кожному із двох заданих речень.

Задача 159. Відредагувати задане речення, видаляючи із нього всі слова з непарними номерами і перевертаючи слова з парними номерами, наприклад, HOW DO YOU DO → OD OD.

Задача 160. Знайти саме довге загальне слово двох заданих речень.

Задача 161. Задано два речення. Знайти саме коротке слово першого речення, якого немає у другому реченні.

Задача 162. Перевірити, чи вірно, що у заданому реченні будь-яке несиметричне слово має парну довжину.

Задача 163. Відредагувати задане речення, видаляючи із нього слова, які зустрічаються в реченні задане число раз.

Задача 164. Надрукувати задане речення таким чином, щоб кожне його слово повністю знаходилося в одному і тому ж рядку роздруківки (тобто уникнути переносів слів та їх розривання).

Задача 165. В заданому реченні вказати слово, в якому частка голосних (A, E, I, O, U) максимальна.

Задача 166. Перелічити всі слова заданого речення, які складаються із тих же букв, що і перше слово речення.

Задача 167. Переставити і роздрукувати слова заданого речення згідно із збільшенням частки приголосних (B, C, D, F, G, H, K, L, M, N, P, Q, R, S, T, V, W, X, Z) у цих словах.

Задача 168. Для кожного символу заданого тексту вказати, скільки разів воно зустрічається у тексті. Повідомлення про один символ повинно друкуватися не більше одного разу.

Задача 169. Замінити закінчення ING кожного слова, що зустрічається у заданому реченні, на ED.

Задача 170. Відредагувати задане речення, видаляючи з нього всі слова, які цілком складені із не більше, ніж двох букв. Приклад: АККА KNOBIKAISE → KNOBIKAISE.

Задача 171. Написати програму, яка у рядку довільної довжини знаходить символ «а» і видаляє за ним 5 символів.

Задача 172. Скласти програму підрахунку загальної кількості цифр і знаків «+», «-», «*» у рядку s , введеного з клавіатури.

Задача 173. Скласти програму підрахунку кількості цифр у заданому рядку.

Задача 174. Скласти програму друку по одному разу в алфавітному порядку всіх великих голосних букв, які входять у заданий текст.

Задача 175. Програма вводить рядок. Якщо довжина рядка < 15 символів, то програма вводить другий рядок і виводить на екран

конкатенацію двох рядків, якщо довжина об'єднаного рядка менше 70-ти символів.

Задача 176. Програма вводить два рядки і виводить менший рядок зеленим кольором, а більший рядок – червоним.

Задача 177. Програма вводить два рядки. Якщо довжина другого рядка більше 50 символів, то програма виводить на екран символи першого рядка з непарними номерами. Якщо довжина першого рядка менше 15 символів, то програма виводить символи другого рядка з парними номерами.

Задача 178. Підрахувати кількість цифр у записі числа n в системі числення з основою k .

Задача 179. Програма вводить два рядки. Вивести на екран, скільки букв «P» у довгому рядку, і скільки букв «R» у короткому рядку. Результат вивести на екран.

Задача 180. Програма вводить два рядки. Вивести на екран конкатенацію рядків, а також вивести кожний рядок на екран, якщо в ньому міститься буквосполучення «окр».

Задача 181. Програма вводить 10 рядків, виводить на екран сполучення 1-го і 10-го рядка, 2-го і 9-го, і т. д.

Задача 182. Програма вводить два рядки довільної довжини і формує масив із номерів позицій, в яких символи рядків не співпадають.

Задача 183. Програма вводить два рядки з клавіатури, виводить на екран конкатенацію 2-х рядків і довжину об'єднаного рядка.

5. КЛАСИ, ПРИНЦИПИ ООП

Задача 184. Описати клас, що реалізовує десятковий лічильник, який може збільшувати або зменшувати своє значення на одиницю в заданому діапазоні. Передбачити ініціалізацію лічильника значеннями за умовчанням і довільними значеннями. Лічильник має два методи: збільшення і зменшення, властивість, що дозволяє отримати його поточний стан. При виході за межі діапазону генерується виключення.

Задача 185. Описати клас, що представляє трикутник. Передбачити методи для створення об'єктів, переміщення на площині, зміни розмірів і обертання на заданий кут. Описати властивості для отримання стану об'єкту. При неможливості побудови трикутника генерується виключення.

Задача 186. Побудувати опис класу, що містить інформацію про поштову адресу організації. Передбачити можливість роздільної зміни складових частин адреси і перевірки допустимості значень, що вводяться. У разі неприпустимих значень полів генерується виключення.

Задача 187. Скласти опис класу для представлення комплексних чисел. Забезпечити виконання операцій складання, віднімання і множення комплексних чисел. Написати програму, що демонструє всі розроблені елементи класу.

Задача 188. Скласти опис класу для вектору, заданого координатами його кінців в тривимірному просторі. Забезпечити операції складання і віднімання векторів з отриманням нового вектору (суми або різниці), обчислення скалярного добутку двох векторів, довжини вектору, косинуса кута між векторами.

Задача 189. Скласти опис класу для представлення дати. Передбачити можливості установки дати і зміни її окремих полів (рік, місяць, день) з перевіркою допустимості значень, що вводяться. У разі неприпустимих значень полів викидаються виключення. Створити методи зміни дати на задану кількість днів, місяців і років.

Задача 190. Скласти опис класу для представлення часу. Передбачити можливості установки часу і зміни його окремих полів

(година, хвилина, секунда) з перевіркою допустимості значень, що вводяться. У разі неприпустимих значень полів генерується виключення.

Задача 191. Скласти опис класу многочленна виду $a*x^2 + b*x + c$. Передбачити методи, що реалізують: обчислення значення многочленна для заданого аргументу; операцію складання, віднімання і множення многочленів з отриманням нового об'єкту –многочлена. Написати програму, що демонструє всі розроблені елементи класу.

Задача 192. Описати клас для роботи з рядком, що дозволяє зберігати тільки двійкове число і виконувати з ним арифметичні операції. Передбачити ініціалізацію з перевіркою допустимості значень. У разі неприпустимих значень генерувати виключення.

Задача 193. Описати клас дробів – раціональних чисел, що є відношенням двох цілих чисел. Передбачити методи складання, віднімання, множення і ділення дробів. Написати програму, що демонструє всі розроблені елементи класу.

Задача 194. Описати клас «процесор», що містить відомості про марку, тактову частоту, об'єм кеша і вартість. Передбачити ініціалізацію з перевіркою допустимості значень полів. У разі неприпустимих значень полів генерувати виключення. Описати властивості для отримання стану об'єкту. Описати клас «материнська плата», що включає клас «процесор» і об'єм встановленої оперативної пам'яті. Передбачити ініціалізацію з перевіркою допустимості значень поля об'єму пам'яті. Описати властивості для отримання стану об'єкту.

Задача 195. Створити базовий клас *Car* (машина), що характеризується торговою маркою (рядок), числом циліндрів, потужністю. Визначити методи перепризначення і зміни потужності. Створити похідний клас *Lorry* (вантажівка), що характеризується також вантажопідйомністю кузова. Визначити функції перепризначення марки та зміни вантажопідйомності.

Задача 196. Створити клас *Liquid* (рідина), що має поля назви і щільності. Визначити методи перепризначення та зміни щільності. Створити похідний клас *Alcohol* (спирт), який має міцність. Визначити методи перепризначення та зміни міцності.

Задача 197. Створити клас *Pair* (пара чисел); визначити методи зміни полів та обчислення добутку чисел. Визначити похідний клас *Rectangle* (прямокутник) з полями-сторонами. Визначити методи обчислення периметра та площі прямокутника.

Задача 198. Створити клас *Man* (людина), з полями: ім'я, вік, стать і вага. Визначити методи перепризначення імені, зміни віку та зміни ваги. Створити похідний клас *Student*, який має поле року навчання. Визначити методи перепризначення та збільшення року навчання.

Задача 199. Створити клас *Triad* (трійка чисел); визначити методи зміни полів і обчислення суми чисел. Визначити похідний клас *Triangle* з полями-сторонами. Визначити методи обчислення кутів та площі трикутника.

Задача 200. Створити клас *Pair* (пара цілих чисел); визначити методи зміни полів і операцію додавання пар $(a, b) + (c, d) = (a + b, c + d)$. Визначити клас-спадкоємець *Long* з полями: старша частина числа і молодша частина числа. Перевизначити операцію додавання та визначити методи множення і віднімання.

Задача 201. Описати клас для роботи з одновимірним масивом цілих чисел (вектором). Забезпечити наступні можливості: задання довільних цілих меж індексів при створенні об'єкту; звернення до окремого елемента масиву з контролем виходу за межі масиву; виконання операцій поелементного складання і віднімання масивів з однаковими межами індексів; виконання операцій множення і ділення всіх елементів масиву на скаляр; виведення на екран елемента масиву по заданому індексу і всього масиву. Написати програму, що демонструє всі розроблені елементи класу.

Задача 202. Описати клас для роботи з вісімковим числом, що зберігається у вигляді рядка символів. Реалізувати конструктори, властивості, методи і наступні операції: операції привласнення, що реалізують значущу семантику; операції порівняння; перетворення в десяткове число; виведення формату; доступ до заданої цифри числа по індексу. Написати програму, що демонструє всі розроблені елементи класу.

Задача 203. Описати клас «студентська група». Передбачити можливість роботи із змінним числом студентів, пошуку студента за якою-небудь ознакою (наприклад, по прізвищу, імені, даті народження), додавання і видалення записів, сортування по різних полях, доступу до запису по номеру.

Задача 204. Описати клас «множина», що дозволяє виконувати основні операції: додавання і видалення елемента, перетин, об'єднання і різниця множин. Написати програму, що демонструє всі розроблені елементи класу.

Задача 205. Описати клас «поїзд», що містить наступні закриті поля: назва пункту призначення; номер поїзда (може містити букви і цифри); час відправлення. Передбачити властивості для отримання стану об'єкту. Описати клас «вокзал», що містить закритий масив поїздів. Забезпечити наступні можливості: виведення інформації про поїзд по номеру за допомогою індексу; виведення інформації про поїзди, що відправляються після введеного з клавіатури часу; перевантажену операцію порівняння, що виконує порівняння часу відправлення двох поїздів; виведення інформації про поїзди, що відправляються в заданий пункт призначення. Інформація має бути відсортована за часом відправлення. Написати програму, що демонструє всі розроблені елементи класів

Задача 206. Описати клас «товар», що містить наступні закриті поля: назва товару; назва магазину, в якому продається товар; вартість товару в гривнях. Передбачити властивості для отримання стану об'єкту. Описати клас «склад», що містить закритий масив товарів. Забезпечити наступні можливості: виведення інформації про товар по номеру за допомогою індексу; виведення на екран інформації про товар, назва якого введена з клавіатури; якщо таких товарів немає, видати відповідне повідомлення; сортування товарів по назві магазину, по найменуванню і за ціною; перевантажену операцію складання товарів, що виконує складання їх цін. Написати програму, що демонструє всі розроблені елементи класів.

Задача 207. Описати клас «англо-російський словник», що забезпечує можливість зберігання декількох варіантів перекладу для

кожного слова. Реалізувати доступ по рядковому індексу - англійському слову. Забезпечити можливість виведення всіх значень слів по заданому префіксу.

Задача 208. Створити клас *Point* (крапка). На його основі створити класи *ColoredPoint* і *Line* (лінія) . На основі класу *Line* створити класи *Coloredline* і *Polyline* (багатокутник) . У класах описати наступні елементи: конструктори з параметрами і конструктори за умовчанням; властивості для установки і набуття значень всіх координат, а також для зміни кольору і отримання поточного кольору; для ліній - методи зміни кута повороту ліній щодо першої крапки; для багатокутника - метод масштабування.

Задача 209. Створити абстрактний клас *Vehicle* (транспортний засіб). На його основі реалізувати класи *Plane* (літак), *Car* (автомобіль) і *Ship* (корабель). Класи повинні мати можливість задавати і отримувати координати і параметри засобів пересування (ціна, швидкість, рік випуску і т. п.) за допомогою властивостей. Для літака має бути визначена висота, для літака і корабля – кількість пасажирів, для корабля – порт приписки. Динамічні характеристики задати за допомогою методів.

Задача 210. Який результат компіляції та виконання наведеного нижче фрагменту коду?

```
static void Main(string[] args)
{
    Console.WriteLine(GetSomeResult(10000));
}
static long GetSomeResult(long someValue)
{
    long value1 = 10 * 1000 * 10000 * someValue;
    long value2 = 10 * 1000 * 10000 * 100000;
    return value2 / value1;
}
```

Задача 211. Яке значення присвоєне *x*, якщо наведений нижче код виводить *False*?

```
float x;
Console.Write(x == x)
```

Задача 212. Пояснити, чому наступний код виводить *False*?

```
public static void Main()
{
    Test t = new Test();
    Console.WriteLine(t.Equals(t));
}
```

Задача 213. Що буде виведено на екран при виконанні наведеного нижче коду?

```
static void Main(string[] args)
{
    char a = 'a';
    int b = 0;
    Console.WriteLine(true ? a : b);
}
```

Задача 214. Що буде виведено на екран при виконанні наведеного нижче коду?

```
NameValueCollection col = new NameValueCollection();
Console.WriteLine("Елемент test " + col["test"] != null ?
    "Існує!" : "Не існує!");
Console.WriteLine("A" + "B" + "C");
Console.WriteLine('A' + 'B' + 'C');
```

Задача 215. Циклічна ініціалізація полів. Який результат користувач побачить на екрані?

```
public class A { public static int x = B.y + 1; }
public class B { public static int y = A.x + 1; }
static void Main(string[] args)
{
    Console.WriteLine("A.x = " + A.x);
    Console.WriteLine("B.y = " + B.y);
}
```

Задача 216. Пояснити хід роботи програми та отриманий результат.

```
int j = 0;
for (int i = 0; i < 10; i++)
    j = j++;
Console.WriteLine(j);
```

Задача 217. Що буде виведено у результаті такого циклу?

```
int end = int.MaxValue;
int begin = end - 100;
```

```
int counter = 0;
for (int i = begin; i <= end; i++)
    counter++;
Console.WriteLine(counter);
```

Задача 218. Який результат користувач побачить на консолі після виконання програми?

```
float begin = 1000000000;
int counter = 0;
for (float i = begin; i < (begin + 10); i++)
    counter++;
Console.WriteLine(counter);
```

Задача 219. Який метод вибере компілятор? Пояснити чому.

```
class A
{
    public void Test(int n) { Console.WriteLine("A"); }
}
class B : A
{
    public void Test(double n) { Console.WriteLine("B"); }
}
static void Main(string[] args)
{
    B b = new B();
    b.Test(5);
}
```

Задача 220. Пояснити хід роботи програми.

```
public class Test
{
    public Test(object obj) { Console.WriteLine("object"); }
    public Test(int[] obj) { Console.WriteLine("int[]"); }
}
public static void Main() { Test t = new Test(null); }
```

Задача 221. Що буде виведено на екран в результаті виконання коду представленого нижче?

```
List<int> list = new List<int>() { 1, 2, 3, 4, 5 };
List<int> all = list.FindAll(i => {
    Console.Write(i);
    return i < 3;});
```

```
List<int> list = new List<int>() { 1, 2, 3 };
var x = list.GroupBy(i => {Console.Write(i); return i; });
var y = list.ToLookup(i => {Console.Write(i); return i; });
```

Задача 222. Що буде виведено на екран у кожному із трьох випадків, наведених нижче?

```
// 1-ий випадок
try {
    Console.WriteLine("Hello ");
    return; }
finally { Console.WriteLine("Goodbye "); }
Console.WriteLine("world!");
```

```
// 2-ий випадок
try {
    Console.WriteLine("Hello");
    Thread.CurrentThread.Abort(); }
finally { Console.WriteLine("Goodbye "); }
Console.WriteLine("world!");
```

```
// 3-ій випадок
try {
    Console.WriteLine("Hello");
    System.Environment.Exit(0); }
finally { Console.WriteLine("Goodbye "); }
Console.WriteLine("world!");
```

6. РОБОТА З ФАЙЛАМИ

Задача 223. Написати програму, яка прочитує з текстового файлу три речення і виводить їх в зворотному порядку.

Задача 224. Написати програму, яка прочитує текст з файлу і виводить на екран тільки речення, що містять введене з клавіатури слово.

Задача 225. Написати програму, яка прочитує текст з файлу і виводить на екран тільки рядки, що містять двозначні числа.

Задача 226. Написати програму, яка прочитує англійський текст з файлу і виводить на екран слова, що починаються з голосних букв.

Задача 227. Написати програму, яка прочитує текст з файлу і виводить його на екран, міняючи місцями кожні два сусідні слова.

Задача 228. Написати програму, яка прочитує текст з файлу і виводить на екран тільки речення, що не містять ком.

Задача 229. Написати програму, яка прочитує текст з файлу і визначає, скільки в нім слів, що складаються не більше ніж з чотирьох букв.

Задача 230. Написати програму, яка прочитує текст з файлу і виводить на екран тільки цитати, тобто пропозиції, заключенні в лапки.

Задача 231. Написати програму, яка прочитує текст з файлу і виводить на екран тільки речення, що складаються із заданої кількості слів.

Задача 232. Написати програму, яка прочитує англійський текст з файлу і виводить на екран слова тексту, що починаються і закінчуються на голосні букви.

Задача 233. Написати програму, яка прочитує текст з файлу і виводить на екран тільки рядки, що не містять двозначних чисел.

Задача 234. Написати програму, яка прочитує текст з файлу і виводить на екран тільки речення, що починаються з тире, перед яким можуть знаходитися тільки пробільні символи.

Задача 235. Написати програму, яка прочитує англійський текст з файлу і виводить його на екран, замінивши прописною кожен першу букву слів, що починаються з голосної букви.

Задача 236. Написати програму, яка прочитує текст з файлу і виводить його на екран, замінивши цифри від 0 до 9 словами «нуль», «один», «дев'ять», починаючи кожне речення з нового рядка.

Задача 237. Написати програму, яка прочитує текст з файлу, знаходить щонайдовше слово і визначає, скільки разів воно зустрілося в тексті.

Задача 238. Написати програму, яка прочитує текст з файлу і виводить на екран спочатку питальні, а потім окличні речення.

Задача 239. Написати програму, яка прочитує текст з файлу і виводить його на екран, після кожного речення додати слово яке введене з клавіатури.

Задача 240. Написати програму, яка прочитує текст з файлу і виводить на екран всі його речення в зворотному порядку.

Задача 241. Написати програму, яка прочитує текст з файлу і виводить на екран спочатку речення, що починаються з однобуквених слів, а потім всі останні.

Задача 242. Написати програму, яка прочитує текст з файлу і виводить на екран речення, яке містить максимальну кількість знаків пунктуації.

Задача 243. Програма запитує шлях до символного файлу і визначає скільки в ньому цифр, скільки в ньому букв латинського алфавіту та скільки символів «?». Результат вивести на екран.

Задача 244. Написати програму, яка запитує ім'я файлу та визначає кількість записаних в нього дійсних чисел більших 7 і менших 21.

Задача 245. Програма запитує назву типізованого файлу і визначає, скільки в ньому дійсних чисел більших 20 або менших 4.

Задача 246. Дано N дійсних чисел. Знайти кількість тих із них, які після відсікання дробової частини кратні 5 або 6, і суму чисел, які більше 4 і їх номер кратний 3. Результат записати у файл, ім'я якого запросити у користувача. Програма запитує N у користувача.

Задача 247. Програма запитує шлях до файлу і виводить на екран компоненти з номерами від 3 до 13, якщо вони є.

Задача 248. Програма запитує шлях до текстового файлу і дописує в нього компонент «С#»

Задача 249. Програма запитує кількість компонентів і створює символний типізований файл, ім'я якого «прізвище.dat». Програма виводить на екран вміст створеного файлу.

Задача 250. Програма запитує ім'я файлу, запитує номер компонента та виводить його на екран, якщо компонент існує.

Задача 251. Програма створює файл, ім'я якого запитує у користувача і заповнює його з клавіатури 10-ю цілими компонентами.

Задача 252. Програма запитує ім'я файлу, запитує компонент і записує його на місце 5-го компонента, якщо число компонентів в файлі більше або рівне 4. Якщо число компонентів менше 4, то програма повідомляє користувача про неможливість зробити запис.

Задача 253. Програма запитує ім'я файлу і номер компонента. Програма замінює компонент із вказаним номером на новий компонент, котрий запитує у користувача, якщо це можливо.

Задача 254. Програма запитує ім'я файлу і номер компонента. Програма виводить заданий компонент на екран у тому випадку, якщо компонент існує. Виділити у окремий метод друк заданого компонента, якщо відомий тип файлу і шлях.

Задача 255. Програма запитує ім'я файлу. Якщо кількість компонентів файлу кратне 3, то вивести на екран суму кожного 3-го компоненту. Якщо кількість компонентів файлу парна, то вивести на екран всі суми кожної пари компонентів.

Задача 256. Виконати сортування типізованого файлу методом бульбашки.

Задача 257. Написати програму, що сортує за зростанням методом бульбашки масив, який вводиться з файлу, ім'я якого повідомляє користувач.

Задача 258. Написати програму, яка запитує ім'я файлу і переписує з нього цілі компоненти через один у другий файл.

Задача 259. Написати програму, яка запитує ім'я текстового файлу і визначає, скільки рядків, записаних у цей файл, мають довжину >15символів і <35символів.

Задача 260. Програма запитує ім'я файлу, в якому записані цілі компоненти. Програма порівнює компоненти попарно. Якщо перший компонент пари більше другого, то в новий файл записати обидва компонента. Якщо перший компонент менше другого, то в новий файл переписати тільки другий компонент.

Задача 261. Програма заповнює випадковими числами цілочисельну матрицю $A[m \times n]$. Запитує цілі числа k і q , виводить на екран елемент, який стоїть у k -ому рядку і q -ому стовпчику матриці A , а також виводить на екран початкову матрицю A . Програма записує перший рядок матриці A у файл `C:\rez.txt`.

Задача 262. Програма запитує ім'я файлу і розміщує в нього 20 записів з 5 полями (ціле, рядок, дійсне, ціле, запис з двома цілими полями).

Задача 263. Програма запитує ім'я текстового файлу і виводить його компоненти на екран.

Задача 264. Поєднати два відсортованих файли в один відсортований файл і вивести результат на екран

Задача 265. У файлі `Array.txt` міститься набір цілих чисел, які записано через крапку з комою. Написати програму, яка обчислює суму та добуток всіх елементів і виводить результати на консоль.

Задача 266.* Написати програму, яка створює файл та генерує послідовність чисел Фібоначчі. Кількість членів ряду користувач задає в консолі. Передбачити введення некоректних даних.

Задача 267.* Написати програму, яка знаходить всі текстові файли у заданому каталозі та поєднує їх (конкатенує). Каталог задається у консолі користувачем. Об'єднувати потрібно тільки файли з розширенням `*.txt`. Результуючий файл назвати `Custom.dat`.

Задача 268.* Написати програму, яка малює за допомогою символів у текстовому файлі графік синуса. Амплітуду синуса задає користувач у консолі. Ім'я вихідного фала – `Image.txt`.

7. ВІЗУАЛЬНЕ ПРОГРАМУВАННЯ

Задача 269. Написати програму, яка малює паркет із прямокутних трикутників заданого розміру.

Задача 270. Написати програму, яка малює паркет із ромбів заданого розміру.

Задача 271. Написати програму, яка малює паркет із парабол заданих параметрів.

Задача 272. Написати програму побудови п'яти графіків заданих функцій.

Задача 273. Побудувати лінії рівня для функції $z = f(x, y)$.

Задача 274. Розробити програму побудови гістограм та діаграм.

Задача 275. Розробити додаток, який дозволяє користувачеві вводити суму грошей у гривнях і отримувати розрахунок в доларах і євро.

Задача 276. Розробити додаток, який дає можливість користувачеві переглядати списки двох груп студентів і здійснювати переведення студентів з однієї групи в іншу за бажанням користувача.

Задача 277. Розробити додаток, який здійснює рух геометричної фігури на формі по горизонталі з відскоком від бічних меж форми. Кожен раз при зіткненні з краєм форми фігура змінює колір.

Задача 278. Розробити додаток, який дає можливість користувачеві вибрати зображення і багаторазово наносити його на форму в різних місцях.

Задача 279. Розробити додаток, який дає можливість користувачеві додавати і видаляти в список групи прізвища студентів за бажанням користувача за умови, що кількість студентів у групі не повинна перевищувати 10 осіб. Результат зберігати у файл.

Задача 280. Розробити додаток, який дає можливість користувачеві з двох груп студентів вивести прізвища, що зустрічаються в обох групах.

Задача 281. Розробити, який дозволяє користувачеві Знайти файл з прізвищами студентів, які успішно пройшли вступні випробування. Сформувані списки груп, чисельність яких не перевищує 5 осіб.

Задача 282. Розробити додаток, який дозволяє вивести на екран найкоротшу прізвище зі списку групи студентів. Список завантажується за бажанням користувача з файлу.

Задача 283. Розробити додаток, який дає можливість користувачеві встановити три довільні фігури і з'єднати їх прямими лініями.

Задача 284. Розробити додаток, який дозволяє, використовуючи список студентів, які не склали іспити, підготувати нові списки груп студентів, переведених на наступний курс.

Задача 285. Розробити додаток, який дозволяє користувачеві багаторазово розміщувати на формі зображення квадрата.

Задача 286. Розробити додаток, який дозволяє користувачеві при правильному введенні пароля, розміщеному на одній формі, створювати іншу форму і здійснювати вертикальне переміщення фігури. При відскоку від верхнього і нижнього краю форми фігура змінює колір.

Задача 287. Розробити додаток, який дає можливість користувачеві за допомогою меню вибрати вид операції та отримувати результат додавання, множення, ділення і віднімання двох дійсних чисел.

Задача 288. Розробити додаток, який дає можливість користувачеві вказати прізвище студента групи і отримати номер рядка списку, в який він записана.

Задача 289. Розробити додаток, який дозволяє управляти рухом «танка» за допомогою клавіш «вправо», «вліво», «вгору», «вниз».

Задача 290. Розробити додаток, який заповнює таблицю елементами текстового файлу, вважаючи таблицю матрицею, дозволяє відредагувати елементи матриці і Знайти середнє арифметичне всіх компонентів матриці. Відредаговану матрицю зберегти у файл. Знайти суму за рядками і за стовпчиками, зберегти результат у додатковому рядку і стовпці таблиці.

Задача 291. Розробити додаток, який призначений для комплектування груп з числа абітурієнтів, які вступають до коледжу. Список абітурієнтів міститься в текстовому файлі. Кількість студентів

у групі – 7 осіб. Проект дозволяє здійснювати переведення студентів з групи в групу і зберігати новий склад групи. Проект повинен мати титульну форму із зазначенням на ній теми і прізвища автора.

Задача 292. Розробити додаток, який здійснює управління пересуванням гармати і стрільбою з неї по мішені. Гармата може рухатися тільки в горизонтальному напрямку, постріл – тільки вертикально вгору. Проект виводить на форму кількість пострілів, які потрібні були для ураження мішені.

Задача 293. Розробити програму, яка малює червоним кольором зображення напівпровідникового діода.

Задача 294.* Розробити електронний калькулятор для виконання операцій $+$, $-$, $*$, $/$ над дійсними числами. Реалізувати можливість розрахунку математичних виразів з операціями в дужках, наприклад, шляхом введення користувачем виразів вигляду $5*((1+2,5)-0,26)$.

Задача 295.* Розробити електронний калькулятор для виконання операцій $+$, $-$, $*$, $/$ над цілими числами у шістнадцятковій системі числення. Реалізувати можливість розрахунку математичних виразів з операціями в дужках, наприклад, шляхом введення користувачем виразів вигляду $F6*(F01C*(A+2)-1B)$.

Задача 296.* Розробити електронний калькулятор для виконання тригонометричних операцій \sin , \cos , tg , ctg над дійсними числами. Реалізувати можливість розрахунку математичних виразів з операціями в дужках, наприклад, шляхом введення користувачем виразів вигляду $\cos(\sin(tg(0.5)))$.

Задача 297.* Розробити програму «Телефонний довідник», яка повинна містити інформацію про прізвище, ім'я абонента, його номер телефону, адресу проживання. Забезпечити можливість збереження та завантаження файлу даних з жорсткого диску. Програма-довідник має дозволяти вносити нові дані до бази та редагувати існуючу інформацію.

Задача 298.* Розробити програму для приховування номерів телефону. Користувач обирає деякий текстовий файл, якщо програма знаходить у ньому номер телефону у форматі $+380XX-XXX-XX-XX$,

або XXX-XX-XX, або XX-XX-XX, то програма замінює три випадкові цифри номеру на зірочки * та зберігає файл під новим ім'ям.

Задача 299.* Розробити графічний редактор, схожий за своїми функціональними можливостями на Windows Paint. Забезпечити наступні функції: малювання ліній, кіл, прямокутників; виведення поточних координат курсора; вибір кольору пера; збереження зображення у файл та завантаження файлу з жорсткого диску.

Задача 300.* Розробити програму «Органайзер», яка відображає поточний час, дату, обсяг вільної оперативної пам'яті. Програма повинна згортатися у трей і при наведенні миші відображати підказкою всю інформацію. При натисненні комбінації Ctrl+Alt+F8 повинен запускатися стандартний калькулятор Windows

ПРИКЛАДИ РОЗВ'ЯЗАННЯ

Приклад 1

Розглянемо програму, яка коротко демонструє застосування основних елементарних конструкцій, циклів та керуючих операторів мови C#.

Розв'язання

```
using System;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            //Hello World
            Console.WriteLine("Hello World!!!");
            Console.WriteLine("Нажмите любую клавишу...");
            Console.ReadKey(); //Ожидание ввода
            //Вывод числовых типов
            int x = 4;
            double y = 20.2;
            Console.WriteLine("У меня было {0} яблока и {1}
рублей", x, y); //Вывод числовых значений
            Console.WriteLine("Нажмите любую клавишу...");
            Console.ReadKey();
            //Вывод строкового типа
            string str = "У меня было 4 яблока и 20.2 рублей";
            Console.WriteLine(str); //Вывод строчного типа
            Console.ReadKey();
            //Объявление одномерного массива
            int[] intArray = new int[10];
            //Объявление двухмерного массива
            int[,] intDoubleArray = new int[10, 10];
            //Инициализация одномерного массива(пример цикла)
            for (int i = 0; i < 10; i++)
            {
                intArray[i] = i;
            }
        }
    }
}
```

```

//Инициализация двухмерного массива
for (int i = 0; i < 10; i++)
{
    for (int j = 0; j < 10; j++)
    {
        intDoubleArray[i, j] = i + j;
    }
}
//Сумма чисел одномерного массива
int count = 0;
int sum = 0;
while (count < intArray.Length)
{
    sum = sum + intArray[count];
    count++;
}
Console.WriteLine("Sum={0}", sum);
Console.WriteLine("Нажмите любую клавишу...");
Console.ReadKey();
}
}
}

```

Приклад 2

Реалізувати програму обчислення суми квадратів двох цілих чисел, які вводить користувач.

Розв'язання

```

using System;
namespace ConsoleApplication
{
    class Program
    {
        static void Main(string[] args)
        {
            string str;
            Console.Write("Введите переменную X: ");
            str = Console.ReadLine();
            int X = Convert.ToInt32(str);
            Console.Write("Введите переменную Y: ");

```



```

        str = Console.ReadLine();
        int Y = Convert.ToInt32(str);
        Console.WriteLine("X^2+Y^2 = {0}", X*X + Y*Y);
        Console.ReadKey();
    }
}

```

Як бачимо, для введення даних користувачем була використана проміжна строкова змінна `string str`; яка потім за допомогою класу `Convert` конвертувалася в ціле число. Таке перетворення є найбільш уразливим місцем у даній програмі, оскільки у випадку введення У випадку букви замість цілого числа виникне помилка та генеруватиметься виняток `System.FormatException`.

Приклад 3

Розробити класи, які б реалізовували малювання геометричних фігур (точки, квадрата) у заданих координатах текстового режиму консолі за допомогою певного символу, наприклад «*». Побудувати ієрархію класів, а базовий клас при цьому об'явити абстрактним, реалізувати механізми наслідування та поліморфізму. У якості поліморфізму виконати перевантаження одного з методів базового класу. Побудувати UML-діаграму класів та діаграму послідовностей засобами Microsoft Visual Studio.

Розв'язання

```

using System;
namespace Lab4_2
{
    class Pen // клас містить властивості олівця, зокрема колір
    {
        public ConsoleColor Color;
        public char Symbol;
        public Pen() // конструктор
        {
            Color = ConsoleColor.White; // білий колір
            Symbol = '*'; // за умовчанням малює зірочками
        }
    }
}

```

```

abstract class Figure
{
    public int x;
    public int y;
    public Pen pen = new Pen();
    public abstract void Draw();
}
class Point : Figure // клас точки
{
    public override void Draw() // перевантаження методу
    {
        Console.ForegroundColor = pen.Color;
        Console.SetCursorPosition(x, y);
        Console.Write(pen.Symbol);
    }
}
class Square : Figure // клас квадрата
{
    public int size = 10; // розмір у символах
    public override void Draw() // перевантаження методу
    {
        int left = x - size / 2;
        int top = y - size / 2;
        for (int i = 0; i <= size; i++)
        {
            Console.ForegroundColor = pen.Color;
            Console.SetCursorPosition(left + i, top);
            Console.Write(pen.Symbol);
            Console.SetCursorPosition(left + i, top+size);
            Console.Write(pen.Symbol);
            Console.SetCursorPosition(left, top + i);
            Console.Write(pen.Symbol);
            Console.SetCursorPosition(left+size, top+i);
            Console.Write(pen.Symbol);
        }
    }
}
class Program
{

```

```

static void Main(string[] args)
{
    Figure fig = new Point();
    fig.x = 10;
    fig.y = 10;
    fig.Draw();
    Figure fig2 = new Square();
    fig2.x = 20;
    fig2.y = 10;
    fig2.Draw(); // приклад поліморфізму
    Console.ReadKey();
}
}
}

```

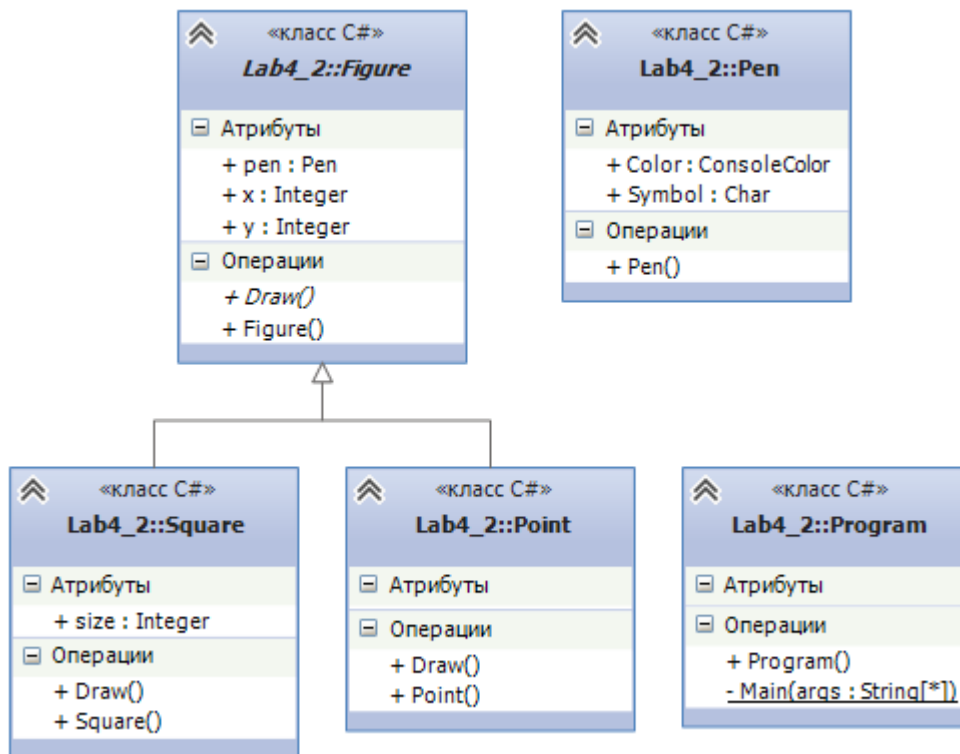


Рисунок 1 – Діаграма класів UML

Для відображення часу життя об'єктів та послідовності викликів методів використовується діаграма послідовностей UML. Для її побудови потрібно обрати фрагмент коду (наприклад, точку входу у програмний додаток) та скористатися контекстним меню редактора коду Microsoft Visual Studio.

У випадку багатопоточних додатків та складній програмній архітектурі такі діаграми можуть бути дуже корисними при документуванні та презентації розроблених програмістом бібліотек (рисунок 2).

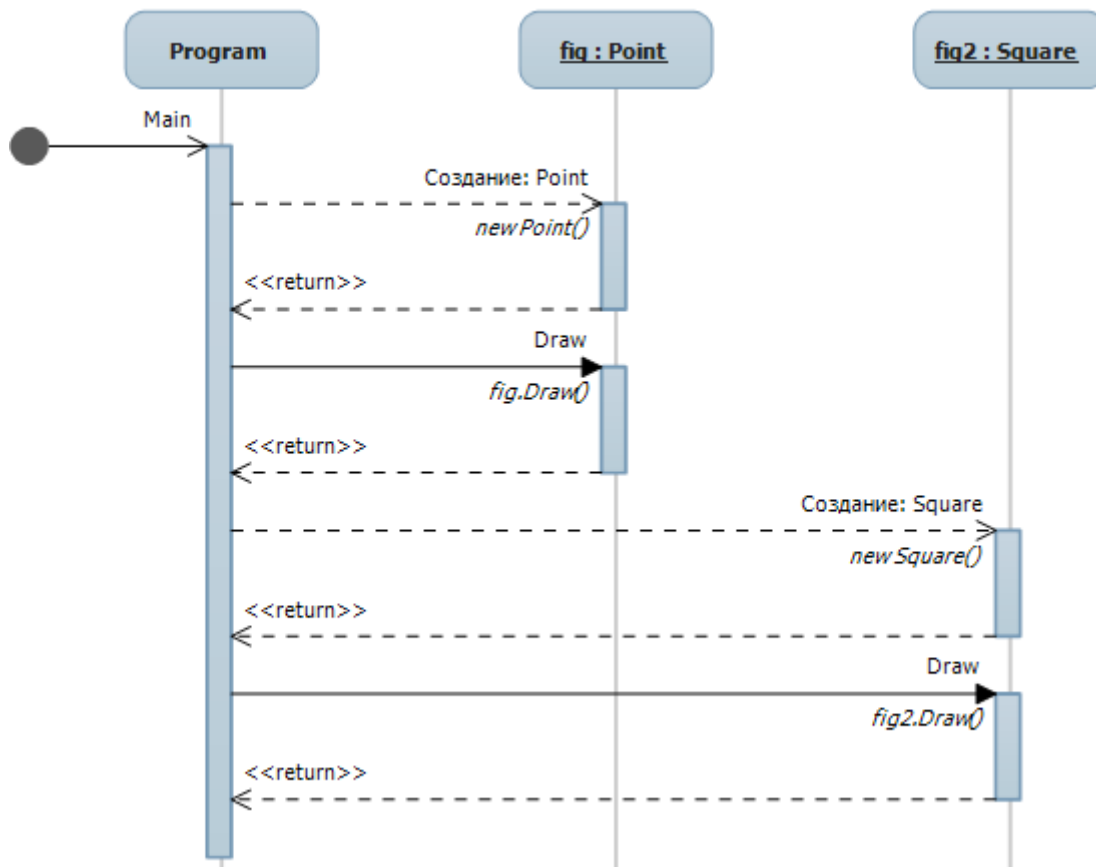


Рисунок 2 – Діаграми послідовностей UML для методу Main

Приклад 4

Реалізувати клас, який інкапсулює властивості матриці розміром 3×3 елемента цілого типу. Реалізувати методи випадкового заповнення, обнулення, транспонування та виведення на консоль. Створити нащадка від базового класу, додавши можливість отримання максимального та мінімального значення в матриці.

Розв'язання

```

using System;
namespace Lab4_3
{
    class Matrix3i // базовий клас матриці
    {
  
```

```

public int[,] m = new int[3, 3];
// формування випадкової матриці, кожний елемент якої
// знаходиться у межах від min до max
public void Random(int min, int max)
{
    Random rnd = new Random();
    for (int i = 0; i < 3; i++)
        for (int j = 0; j < 3; j++)
            m[i, j] = rnd.Next(min, max);
}
public void Print() // друк на консоль
{
    Console.WriteLine("\nm = ");
    Console.WriteLine("{0}\t{1}\t{2}", m[0,0], m[0,1],
m[0, 2]);
    Console.WriteLine("{0}\t{1}\t{2}", m[1,0], m[1,1],
m[1, 2]);
    Console.WriteLine("{0}\t{1}\t{2}", m[2,0], m[2,1],
m[2, 2]);
}
public void Transpose() // транспонування матриці
{
    int tmp;
    tmp = m[0, 1]; m[0, 1] = m[1, 0]; m[1, 0] = tmp;
    tmp = m[0, 2]; m[0, 2] = m[2, 0]; m[2, 0] = tmp;
    tmp = m[1, 2]; m[1, 2] = m[2, 1]; m[2, 1] = tmp;
}
public void SetZero() // заповнення нулями
{
    for (int i = 0; i < 3; i++)
        for (int j = 0; j < 3; j++)
            m[i, j] = 0;
}
public Matrix3i() // конструктор
{
    Console.WriteLine("Create Matrix3i");
}
}

```

```

class Matrix3iEx : Matrix3i // похідний клас
{
    public int MaxValue // властивість тільки для читання
    {
        get
        {
            int max = m[0, 0];
            for (int i = 0; i < 3; i++)
                for (int j = 0; j < 3; j++)
                    if (m[i, j] > max)
                        max = m[i, j];
            return max;
        }
    }
    public int MinValue // властивість тільки для читання
    {
        get
        {
            int min = m[0, 0];
            for (int i = 0; i < 3; i++)
                for (int j = 0; j < 3; j++)
                    if (m[i, j] < min)
                        min = m[i, j];
            return min;
        }
    }
    public Matrix3iEx() // конструктор
    {
        Console.WriteLine("Create Matrix3iEx");
    }
}
class Program
{
    static void Main(string[] args)
    {
        Matrix3iEx m = new Matrix3iEx();
        m.Print();
        m.Random(-10, 10);
        m.Print();
    }
}

```

```

        m.Transpose();
        m.Print();
        Console.WriteLine();
        Console.WriteLine("Max = {0}", m.MaxValue);
        Console.WriteLine("Min = {0}", m.MinValue);
        Console.ReadKey();
    }
}
}

```

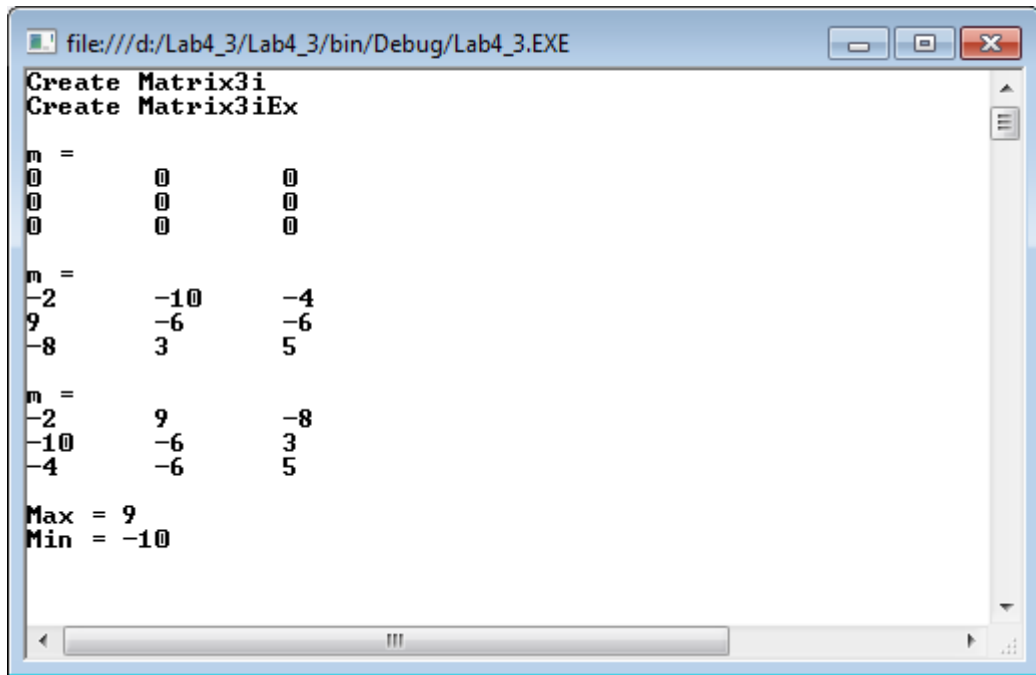


Рисунок 3 – Демонстрація роботи програми

Приклад 5

Дано список цілих чисел, який складається з n елементів. Обчислити суму та кількість чисел, що діляться без остачі на 10. Якщо таких чисел немає, вивести повідомлення «Чисел, що діляться без остачі на 10, у списку немає».

Розв'язання

Один із варіантів вирішення даного завдання представлено нижче:

```

using System;
using System.Collections.Generic;
namespace Collection {
    class Program
    {
        const int n = 50;

```

```

static void Main(string[] args)
{
    List<int> list = new List<int>();
    Random rnd = new Random();
    for (int i = 0; i < n; i++)
    {
        list.Add(rnd.Next(100));
    }
    int sum = 0;
    int count = 0;
    Console.Write("List: ");
    foreach (int a in list)
    {
        Console.Write("{0}; ", a);
        if (a % 10 == 0)
        {
            count++;
            sum += a;
        }
    }
    Console.WriteLine();
    if (count == 0)
    {
        Console.WriteLine("Чисел, що діляться без остачі на
                            10, у списку немає");
    }
    else
    {
        Console.WriteLine("Count: {0}\nSum: {1}", count,
                            sum);
    }
    Console.ReadKey();
}
}
}

```



```
file:///D:/Collection/Collection/bin/Debug/Collection.EXE
List: 14; 28; 53; 60; 50; 95; 24; 30; 77; 98; 12; 4; 89; 51; 83; 84; 72; 70; 4;
82; 4; 14; 64; 65; 80; 95; 75; 69; 72; 43; 33; 81; 69; 58; 3; 48; 43; 6; 41; 41;
91; 4; 41; 27; 20; 5; 3; 22; 14; 93;
Count: 6
Sum: 310
```

Рисунок 4 – Результат роботи програми

Приклад 6

Виконати читання текстового файлу та знайти суму чисел у кожному рядку.

Розв'язання

На вміст файлу накладаються строгі обмеження: числа мають бути розділені рівно одним пропуском, після останнього числа в рядку пропуску бути не повинно, файл не повинен закінчуватися символом переведення рядків.

```
using System;
using System.IO;
namespace FileTest
{
    class Class1
    {
        static void Main(string[] args)
        {
            try
            {
                StreamReader f = new StreamReader("1.txt");
                string s;
                const int n = 20;
                int[] a = new int[n];
                string[] buf;
                while ((s = f.ReadLine()) != null)
                {
                    buf = s.Split(' ');
                    long sum = 0;
                    for (int i = 0; i < buf.Length; ++i)
                    {
                        a[i] = Convert.ToInt32(buf[i]);
                    }
                }
            }
        }
    }
}
```

```

        sum += a[i];
    }
    Console.WriteLine("{0} сума: {1}",s, sum);
}
f.Close();
}
catch (FileNotFoundException e)
{
    Console.WriteLine(e.Message);
    Console.WriteLine(" Перевірте ім'я файлу!");
    return;
}
catch (Exception e)
{
    Console.WriteLine("Error; " + e.Message);
    return;
}
}
}
}

```

Результат роботи програми:

```

1 2 4 сума: 7
3 44 -3 6 сума: 50
8 1 1 сума: 10

```

Приклад 7

Створити клас, який представляє комплексне число. Реалізувати перевантаження основних арифметичних операцій (+, -, *, /). Продемонструвати роботу даних операцій.

Розв'язання

Як відомо з курсу математики, комплексне число має дійсну та уявну частину.

```

class Complex
{
    public double Re; // real part
    public double Im; // imagine part
    public Complex() { }
    public Complex(double re, double im)

```

```

    {
        Re = re;
        Im = im;
    }
    public static Complex operator +(Complex a, Complex b)
    {
        return new Complex(a.Re + b.Re, a.Im + b.Im);
    }
    public static Complex operator -(Complex a, Complex b)
    {
        return new Complex(a.Re - b.Re, a.Im - b.Im);
    }
    public static Complex operator *(Complex a, Complex b)
    {
        return new Complex(a.Re * b.Re - a.Im * b.Im,
                            a.Re * b.Im + a.Im * b.Re);
    }
    public static Complex operator /(Complex a, Complex b)
    {
        double val = 1 / (b.Re * b.Re - b.Im * b.Im);
        return new Complex(val * (a.Re * b.Re + a.Im * b.Im),
                            val * (a.Re * b.Im - a.Im * b.Re));
    }
    public override string ToString()
    {
        return String.Format("{0} + {1}*j", Re, Im);
    }
}
class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Complex class testing");
        Complex a = new Complex(3, 4);
        Complex b = new Complex(2, -3);
        Complex c;
        Console.WriteLine("a = {0}", a);
        Console.WriteLine("b = {0}", b);
        c = a + b;
    }
}

```

```

    Console.WriteLine("c = a + b = {0}", c);
    c = a - b;
    Console.WriteLine("c = a - b = {0}", c);
    c = a * b;
    Console.WriteLine("c = a * b = {0}", c);
    c = a / b;
    Console.WriteLine("c = a / b = {0}", c);
    Console.ReadKey();
}
}

```

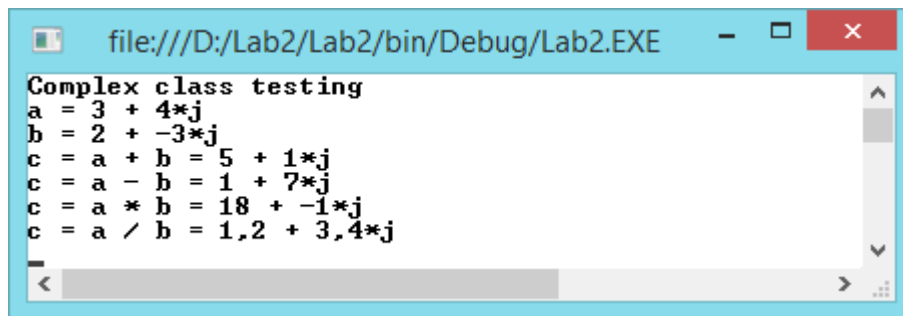


Рисунок 5 – Приклад виконання програми

Приклад 8

За допомогою методів розширення додати до типу `int` можливість зведення його в квадрат та куб.

Розв'язання

```

static class IntEx
{
    public static int Sqr(this int val)
    {
        return val * val;
    }
    public static int Cube(this int val)
    {
        return val * val * val;
    }
}
class Program
{
    static void Main(string[] args)
    {

```

```

    Console.WriteLine(25.Sqrt());
    Console.WriteLine((5 + 3).Cube());
    Console.ReadKey();
}
}

```

Приклад 9

Створити додаток Windows Forms. Обчислити значення виразу

$$\sum_{i=1}^N \frac{a + i^2 - \sqrt{b}}{\sin(i\pi) + \sqrt{\pi}},$$

передбачивши обробку помилок та видачу відповідних повідомлень користувачу. Параметри виразу a , b , N користувач вводить у вікні програми.

Розв'язання

На формі необхідно налаштувати такі візуальні компоненти (рисунок 6).

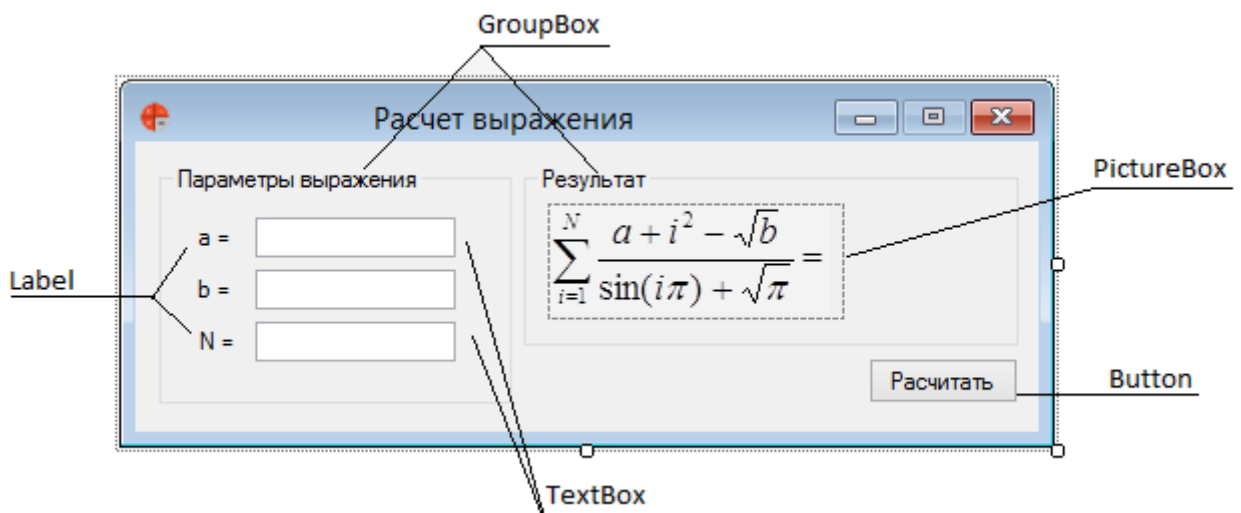


Рисунок 6 – Створення форми у редакторі Form Design

Для форми доцільно налаштувати такі властивості:

Властивість	Значення	Пояснення
Name	FormMain	назва форми
FormBorderStyle	FixedSingle	заборонити можливість зміни розміру вікна
Icon	Імпортоване зображення	іконка у верхньому лівому кутку вікна
MaximizeBox	False	заборона розгортати вікно на весь екран

Властивість	Значення	Пояснення
StartPosition	CenterScreen	запускати вікно посередині екрану
Text	Расчет выражения	назва, що відображається у заголовку вікна

Нижче подано частину програмного коду форми.

```
public partial class FormMain : Form
{
    public FormMain()
    {
        InitializeComponent();
    }
    // Обработка ввода только действительных чисел
    private void textBoxA_KeyPress(object sender,
                                   KeyPressEventArgs e)
    {
        TextBox textBox = (TextBox)sender;
        if (char.IsControl(e.KeyChar))
            return;
        if (e.KeyChar.Equals('-') && textBox.Text.Length == 0)
            return;
        if (e.KeyChar.Equals('.') && !textBox.Text.Contains('.'))
            return;
        if (char.IsDigit(e.KeyChar))
            return;
        e.Handled = true;
    }
    private void buttonCalc_Click(object sender, EventArgs e)
    {
        try
        {
            double a = double.Parse(textBoxA.Text),
                b = double.Parse(textBoxB.Text),
                N = double.Parse(textBoxN.Text),
                S = 0;
            for (int i = 0; i < N; i++)
            {
```

```

        S += (a + i * i - Math.Sqrt(b)) /
            (Math.Sin(i * Math.PI) + Math.Sqrt(Math.PI));
    }
    labelRes.Text = String.Format("{0:f3}", S);
}
catch (FormatException)
{
    MessageBox.Show("Проверьте параметры", "Ошибка");
}
catch (DivideByZeroException)
{
    MessageBox.Show("Деление на ноль :(", "Ошибка");
}
}

```

Якщо користувач не вводить ніякі дані, то виникає помилка (рисунок 7)

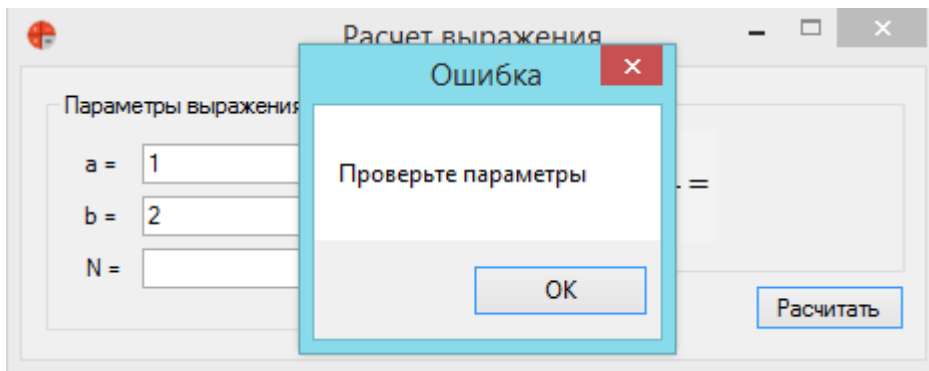


Рисунок 7 – Помилка роботи додатку

Приклад 10

Розробити програму типу Windows Forms Application та програмно намалювати червоним кольором зображення напівпровідникового діода.

Розв'язання

Зважаючи на необхідність звільнення пам'яті при використанні класу Pen, програмний код може мати наступний вигляд:

```

public partial class FormMain : Form
{
    Pen pen;
    public FormMain()
    {

```

```

InitializeComponent();
pen = new Pen(Color.Red, 3);
}
private void FormMain_Paint(object sender, PaintEventArgs e)
{
    e.Graphics.DrawLine(pen, 50, 50, 80, 50);
    e.Graphics.DrawPolygon(pen, new Point[] {
        new Point(80, 30),
        new Point(80, 70),
        new Point(120, 50) });
    e.Graphics.DrawLine(pen, 120, 30, 120, 70);
    e.Graphics.DrawLine(pen, 120, 50, 150, 50);
}
private void FormMain_FormClosing(object sender,
                                   FormClosingEventArgs e)
{
    pen.Dispose();
}
}

```

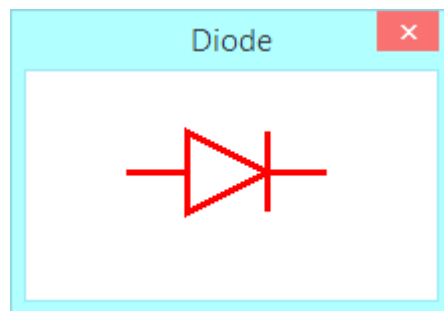


Рисунок 8 – Результат роботи програми

ТЕМИ РЕФЕРАТІВ

1. Об'єктно-орієнтовані мови програмування.
2. Основні особливості мови Object Pascal.
3. Принципи ООП на прикладі мови Java.
4. Огляд об'єктно-орієнтованих засобів мови C++.
5. Мова PHP: синтаксис, класи, основні бібліотеки.
6. JavaScript та принципи ООП.
7. Уніфікована мова моделювання UML.
8. Принципи ООП (інкапсуляція, спадкоємство, поліморфізм).
9. Архітектура платформи Microsoft .NET.
10. Середовище розробки додатків Microsoft Visual Studio.
11. Основні класи бібліотеки .NET Framework.
12. Системні типи даних мови C#.
13. Поняття інтерфейсу та узагальнення.
14. Клас в об'єктно-орієнтованих мовах програмування.
15. Введення-виведення інформації у файл, серіалізація об'єктів.
16. Засоби розробки програмного забезпечення для мобільних пристроїв.
17. Порівняльна характеристика інтегрованих середовищ розробки IDE.
18. Шаблони проектування (Design Patterns).
19. Огляд найбільш розповсюджених фреймворків (Framework).
20. Коротка характеристика технологій Windows Forms та Windows Presentation Foundation.
21. Технології модульного тестування та принципи зменшення зв'язності модулів.
22. Віртуальна машина, інтерпретовані мови програмування.
23. Засоби створення кросплатформових додатків.
24. Аналіз ринку праці у галузі програмування.
25. Головні тенденції об'єктно-орієнтованого проектування інформаційних систем.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

Основна література

1. Троелсен Э., Джепикс Ф. Язык программирования C# 7, платформы .NET и .NET Core : 8-е изд., пер. с англ. СПб. : Диалектика, 2018. 1328 с.
2. Скит Джон. C# для профессионалов. Тонкости программирования. М. : Вильямс, 2019. 608 с.
3. Тузовский А. Ф. Объектно-ориентированное программирование. М. : Юрайт, 2018. 206 с.
4. Албахари Джозеф, Албахари Бен. C# 7.0. Справочник. Полное описание языка 7-е изд. М. : Вильямс, 2018. 1220 с.
5. Рихтер Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# : 4-е изд. СПб. : Питер, 2018. 896 с.
6. Васильев А. Программирование на C# для начинающих. Основные сведения. М. : Эксмо, 2018. 592 с.
7. Тепляков С. Паттерны проектирования на платформе .NET. СПб. : Питер, 2015. 320 с.

Додаткова література

8. Troelsen Andrew, Japikse Phil. Pro C# 8 with .NET Core 3. Foundational Principles and Practices in Programming : 9th ed. USA : Apress, 2020. 1262 p.
9. Jon Skeet. C# in Depth : 4th ed. USA : Manning Publications, 2019. 528 p.
10. John Sharp. Microsoft Visual C# Step by Step. USA : Microsoft Press, 2018. 832 p.
11. Nagel Christian. Professional C# 7 and .NET Core 2.0. Birmingham : Wrox, 2018. 1440 p.
12. Solis Daniel, Schrottenboer Cal. Illustrated C# 7. The C# Language Presented Clearly, Concisely, and Visually : 5th ed. USA : Apress, 2018. 817 p.

13. Perkins Benjamin, Jacob Vibe Hammer, Jon D. Reid. *Beginning C# 7 Programming with Visual Studio 2017*. Birmingham : Wrox, 2018. 912 p.
14. Sarcar Vaskaran. *Design Patterns in C#. A Hands-on Guide with Real-World Examples*. USA : Apress, 2018. 465 p.
15. MacDonald M. *Pro .NET 2.0 Windows Forms and Custom Controls in C#*. USA : Apress, 2006. 1081 p.
16. Албахари, Джозеф, Албахари, Бен. *C# 8.0. Карманный справочник*. СПб. : Диалектика, 2020. 240 с.
17. Прайс М. Дж. *C# 7 и .NET Core. Кросс-платформенная разработка для профессионалов : 3-е изд.* СПб. : Питер, 2018. 640 с.
18. Вагнер Билл. *Эффективное программирование на C#. 50 способов улучшения кода*. М. : Вильямс, 2017. 224 с.
19. Бойко Б. І., Омельчук Л. Л., Русіна Н. Г. *Об'єктно-орієнтоване програмування. Лабораторний практикум : навч. посіб.* К. : Київський національний університет ім. Тараса Шевченка, 2016. 90 с.
20. Гаско Рик. *Объектно-ориентированное программирование*. М. : Солон-Пресс, 2016. 298 с.
21. Фленов М. Е. *Библия C# : 3-е изд., перераб. и доп.* СПб. : БХВ-Петербург, 2016. 544 с.
22. Бублик В. В. *Об'єктно-орієнтоване програмування : підручник* К. : ІТ-книга, 2015. 624 с.
23. Петцольд Ч. *Программирование с использованием Microsoft Windows Forms. Мастер-класс*. М. : Русская Редакция; СПб. : Питер, 2006. 432 с.

ЗМІСТ

Вступ.....	3
Варіанти індивідуальних завдань	4
1. Умовні оператори, цикли	6
2. Одновимірні масиви.....	13
3. Двовимірні матриці	16
4. Символьні рядки.....	19
5. Класи, принципи ООП.....	23
6. Робота з файлами.....	31
7. Візуальне програмування	35
Приклади розв'язання	39
Теми рефератів.....	57
Список рекомендованої літератури	58

Методичні вказівки
до виконання самостійних робіт
з дисципліни «Об'єктно-орієнтоване програмування»
для студентів спеціальності
123 «Комп'ютерна інженерія»
усіх форм навчання

УКЛАДАЧІ: Музика Іван Олегович

Кузнєцов Денис Іванович

Реєстраційний № _____

Підписано до друку _____ 2021 р.

Формат _____ А5 _____

Обсяг _____ 61 _____ стор.

Тираж _____ прим.

Видавничий центр
Криворізького національного університету,
вул. Віталія Матусевича, 11, м. Кривий Ріг