

1. Простий клас Java. Командний рядок. Методи та функції

Аудиторна робота

- 1.1. Створіть клас, що містить неініціалізовані поля типів `int`, `char` та Рядок, і надрукуйте їх значення, щоб переконатися, що Java виконує ініціалізацію за замовчуванням.
- 1.2. Напишіть програму "hello, world".
- 1.3. Скопіюйте програму з `javac` і запустіть її за допомогою `java`. Якщо ви використовуєте інше середовище розробки, ніж `JDK`, дізнайтеся, як компілювати та запускати програми в цьому середовищі.
- 1.4. Напишіть програму, яка демонструє, що незалежно від того, скільки об'єктів ви створюєте для певного класу, у цьому класі є лише один екземпляр певного статичного поля.

Командний рядок

- 1.5. Напишіть програму, яка друкує три аргументи, взяті з командного рядка.
- 1.6. Знайти суму аргументів командного рядка та порахувати введені дійсні числа.
- 1.7. Ввести ім'я користувача за допомогою командного рядка
- 1.8. Задokumentуйте правильно програму. Виконайте `Javadoc` для файлу та перегляньте результати за допомогою веб-браузера. Перевірте отриману документацію за допомогою веб-браузера.
- 1.9. Додайте список елементів HTML до документації. Візьміть програму попередньої вправи і додайте до неї документацію з коментарями. Розпакуйте цю документацію з коментарями у файл HTML за допомогою `Javadoc` і перегляньте її у веб-браузері.

В приведених завданнях потрібно зчитати дані з консолі та вивести відповідь в консоль. В консолі також вивести прізвище студенту, час отримання завдання та його номер. Додати коментар на початку в програми у вигляді `/**коментар*/` та згенерувати html-файл документації.

- 2.1. Вивести в консоль підказку «Введіть прізвище», введіть його з консолі та виведіть повідомлення «Привіт, ***» де замість зірочок — введене прізвище.

- 2.2. Введіть два цілих числа та виведіть кількість їх розрядів (розв'яжить це за допомогою рядків та використовуючі логарифм). Обчисліть середнє гармонічне цих чисел та виведіть його з точністю до двох знаків після коми.
- 2.3. Ввести два дійсних числа записаних через пробіли та обчислити значення їх середнього геометричного. Результат вивести в науковому та десятковому представленні.
- 2.4. Ввести дійсне число градусів Цельсія C (на екрані повинна бути підказка, що ввести) та обчислити й вивести число F в дійсному форматі – та сама температура в градусах Фаренгейта за формулою $F = 9 * C / 5 + 32$. Результат вивести в окремому рядку вигляду «F=*** », де замість зірок представлення в найкоротшому вигляді з можливих.
- 2.5. Запустити програму з командного рядку. Відобразити в консолі аргументи командного рядку в зворотньому порядку.
- 2.6. Вивести задану кількість випадкових чисел з переходом і без переходу на новий рядок.
- 2.7. Ввести цілі числа як аргументи командного рядку, підрахувати їх добуток та вивести результат на консоль.

Прості типи та оператори

- 3.1. Обчисліть наступні математичні вирази та виведіть результати:
 $1.2+31$; $45*54-11$; $15/4$; $15.0/4$; $67\%5$; $(2*45.1 + 3.2)/2$;
- 3.2. Обчисліть результати наступних виразів та вивести на екран напис українською мовою «Результат дорівнює:»:
 $2+3$; $4.5*56$; $2/3.0$.
- 3.3. Виведіть напис : «Введіть ім'я:»
Введіть з нового рядка ваше ім'я (наприклад, «Вася») та виведіть привітання вигляду «Привіт, Вася!»
- 3.4. Ініціалізуйте наступні числа як дійсні, подвійні дійсні та довгі дійсні: 10^{-4} , $2.33E5$, π , e , $qrt5$, $ln(100)$
- 3.5. Задайте в програмі довільні 5 цілих та 5 дійсних чисел. Вивести на екран таблицю з цих значень у вигляді, слідкуючі за "красою" виводу:

x		1		2		3		4		5	
-		-	-		-	-		-	-		-
y		3.0		1.0		5.0		4.0		2.1	

- 3.6. Ініціалізувати два довільні рядки та вивести їх в одному рядку та поставивши між ними кому та пробіл, а перед та після три окличних знаки. Приклад:
!!! Hello , World! !!!
- 3.7. Обчислити силу притягання F в науковому (експоненційному) форматі між двома тілами, що мають маси m_1, m_2 на відстані r . Вказівка. Шукана сила визначається за формулою $F = \gamma \frac{m_1 * m_2}{r^2}$, де $\gamma = 6.673 * 10^{-11} \text{ Н} * \text{м}^2 / \text{кг}^2$. Всі потрібні змінні присвоюються всередині програми. Результат вивести в окремому рядку вигляду «F=*** », де замість зірок представлення в науковому (експоненційному) вигляді.
- 3.8. Ввести дійсне число x та підрахуйте без та за допомогою математичних функцій її цілу та дробову частину, найменше ціле число, що більше x та найбільше ціле, що менше x , а також його округлене значення. Перевірте результат роботи для від'ємного числа.
- 3.9. Ввести в двох різних рядках послідовно два дійсних числа та обчислити значення їх різниці та добутку. Результат вивести в десятковому представленні (з фіксованою крапкою).
- 3.10. Ввести два дійсних числа записаних через пробіли в одному рядку та обчислити значення їх середнього арифметичного та середнього гармонічного. Результат вивести в науковому та десятковому представленні.
- 3.11. Три дійсні числа вводяться як рядок вигляду
A=xxx.xxx, B=xxExxx C=xxx.xxxx , де "A=","B=","C=" символи, що повинні бути присутніми та ігноруються при введенні (Бажано не використовувати рядковий тип при введенні).
- 3.12. Обчисліть їх середнє арифметичне та середнє гармонічне та виведіть у науковому та форматі з фіксованою крапкою.
- 3.13. Ввести дійсне число від 0 до 10000 та вивести його 8 ступінь з точністю до 20 знаків до десяткової коми та 4 значками після десяткової коми.
- 3.14. На терміналі вводяться 20 цифр. Перші 10 цифр – це перше ціле число, останні 10 цифр – друге. Введіть їх (не використовуючі рядковий тип) та обчисліть і виведіть їх суму.
- 3.15. Позиція у грі «Хрестики-нулики» представлена в програмі за допомогою 9 символів виду ' ','O','X'. Показати на екрані позицію у грі «Хрестики-нулики». Наприклад, для позицій { 'O','X',' ',' ',' ','X','O','X','O',' ',' ' } вона буде:
O | X |
| X | O

X | O |

- 3.16. Ввести користуючись лише однією функцією вводу ціле число записане в шістнадцятковому вигляді та вивести його зменшене на одиницю в шістнадцятковому та десятковому вигляді.
- 3.17. Дійсне число записано в рядку, при цьому перед ним може стояти будь-яка послідовність з пробілів та символів '*'. Ввести його користуючись лише одним викликом функції вводу та виведіть значення його кубу.
- 3.18. Введіть два цілих числа n, m та виведіть числа m, m^2 в різних рядках на відстанях від лівого краю консолі рівних n та $2n$ відповідно.
- 3.19. Ввести дійсне число x та обчислити значення функції тригонометричного косинуса для нього.
- 3.20. Обчислити гіпотенузу c прямокутного трикутника за катетами a та b .
- 3.21. Обчислити площу трикутника S за трьома сторонами a, b, c .
- 3.22. В трикутнику відомо довжини всіх сторін. Обчислити довжини його:
 - а) медіан,
 - б) бісектрис,
 - в) висот.
- 3.23. Трикутник заданий величинами своїх кутів та радіусом вписаного кола. Обчисліть його площу.
- 3.24. Трикутник заданий довжиною своїх сторін. Знайти та вивести величину кутів трикутника у радіанах та градусах.
- 3.25. Обчислити відстань від точки (x_0, y_0) до:
 - а) заданої точки (x, y) ;
 - б) заданої прямої $ax + by + c = 0$;
 - в) точки перетину прямих $x + by + c = 0$ і $ax + y + c = 0$, де $ab \neq 1$.
- 3.26. Знайти об'єм циліндра, якщо відомо його радіус основи та висоту.
- 3.27. Знайти об'єм конуса, якщо відомо його радіус основи та висоту.
- 3.28. Знайти об'єм тора з внутрішнім радіусом r і зовнішнім радіусом R .
- 3.29. Знайти корені квадратного рівняння з коефіцієнтами a, b, c , якщо відомо, що обидва корені в ньому існують. Перевірте ваш розв'язок на коефіцієнтах рівняння $a=3, b=100, c=2$.

Декларація та використання функцій

- 4.1. Напишіть функцію, яка за найменшу кількість арифметичних операцій, обчислює значення многочлена для введеного з клавіатури значення x :

а) $y = x^4 + 2x^2 + 1$;

б) $y = x^4 + x^3 + x^2 + x + 1$;

в) $y = x^5 + 5x^4 + 10x^3 + 10x^2 + 5x + 1$;

г) $y = x^9 + x^3 + 1$;

д) $y = 16x^4 + 8x^3 + 4x^2 + 2x + 1$;

е) $y = x^5 + x^3 + x$.

- 4.2. Скласти функцію для обчислення значення многочлена від двох змінних для введеної з клавіатури пари чисел (x, y) :

а) $f(x, y) = x^3 + 3x^2y + 3xy^2 + y^3$;

б) $f(x, y) = x^2y^2 + x^3y^3 + x^4y^4$;

в) $f(x, y) = x + y + x^2 + y^2 + x^3 + y^3 + x^4 + y^4$.

- 4.3. Напишіть функцію $Rosenbrock2d(x, y) = 100(x^2 - y)^2 + (x - 1)^2$ та перевірте її результат на довільних трьох парах дійсних чисел.

- 4.4. Трикутник заданий довжинами своїх сторін. Знайти периметр та площу цього трикутника. Перевірте для значень сторін $a = 3, b = c = 3.5 + 3 * 2^{-111}$

- 4.5. Трикутник вводиться координатами своїх вершин, які вводяться так: в першому рядку через пробіл два дійсних числа — координати точки А, пропускається рядок, в третьому рядку через пробіл два дійсних числа — координати Б, пропускається рядок, через пробіл — координати точки С. Підрахувати площу трикутника. (Вказівка: напишіть функції підрахунку довжини відрізка та функції обчислення площі трикутника за довжинами сторін)

- 4.6. Напишіть власні функції, що обчислюють наступні вирази та відповідні власні функції, що будуть рахувати похідні даних функцій (Приклад, функція $f(x) = identity(x) = x$, її похідна $g(x) = identity_derivative(x) = 1$) :

а) $f(x) = th(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$;

б) $f(x) = bent(x) = \frac{\sqrt{x^2 + 1} - 1}{2} + x$;

в) $f(x) = softSign(x) = \frac{x}{1 + |x|}$;

г) $f(x) = arctg(x) = tg^{-1}(x)$;

- д) $f(x) = gauss(x) = e^{-x^2}$;
- е) $f(x) = softPlus(x) = \ln(1 + e^x)$;
- є) $f(x) = sigmoid(x) = (1 + e^{-x})^{-1}$;
- ж) $f(x) = invsqrt(x, \alpha) = \frac{x}{\sqrt{1 + \alpha x^2}}$;
- з) $f(x) = sigmweight(x) = x * (1 + e^{-x})^{-1}$.

2. Оператори та типи даних Java. Умовні конструкції. Цикли

Прості типи

- 1.1. Напишіть програму Java для відображення значення за замовчуванням усіх примітивних типів даних Java.
- 1.2. Напишіть програму перевірки, що два рядки рівні чи ні
- 1.3. Напишіть програму, яка використовує `print` та `printf` форму виведення.
- 1.4. Створіть клас, що містить `float`, і використовуйте його для демонстрації псевдонімів
- 1.5. Створіть клас, що містить `float`, і використовуйте його для демонстрації псевдонімів під час викликів методів.
- 1.6. Створіть клас під назвою `Dog`, що містить два рядки: `name` і `say`. У `main ()` створіть два собачі об'єкти з іменами "spot" (хто каже: "Ruff!") та "scruffy" (хто каже: "Wurf!"). Потім покажіть їхні імена та те, що вони говорять. Після вправи 5 створіть нове посилання на собаку та призначте її об'єкту "spot". Перевірте порівняння за допомогою `==` та `equals ()` для всіх посилань.
- 1.7. Напишіть програму, яка імітує підкидання монет.
- 1.8. Покажіть, що шістнадцяткові та вісімкові визначення працюють з довгими значеннями. Використовуйте `Long.toBinaryString ()` для відображення результатів.
- 1.9. Напишіть метод, який приймає два аргументи `String` і використовує всі булеві порівняння для порівняння двох рядків і друкує результати. Для `==` та `!=` Також виконайте тест `equals ()`. У `main ()` викликайте свій метод з різними об'єктами `String`.

Бітові операції

- 2.1. Напишіть програму з двома постійними значеннями, одне з змінними двійковими та нулями, з нулем у найменш значущій цифрі, а друге, також поперемінним, з одиницею у найменш значущій цифрі (підказка: Найпростіше використовувати шістнадцяткову систему константи для цього). Візьміть ці два значення та об'єднайте їх усіма можливими способами, використовуючи побітові оператори, і виведіть результати за допомогою `Integer.toBinaryString()`. Почніть з числа, яке має найбільшу значущу двійкову позицію (підказка: Використовуйте шістнадцяткову константу).
- 2.2. Використовуючи знаковий оператор зсуву вправо, змістіть його вправо до кінця через усі його двійкові позиції, кожен раз відображаючи результат за допомогою `Integer.toBinaryString()`. Почніть з числа, яке є двійковим. Зсуньте його вліво, а потім використовуйте безпідписаний оператор зсуву вправо для переміщення вправо через усі його двійкові позиції, кожен раз відображаючи результат за допомогою `Integer.toBinaryString()`.
- 2.3. Напишіть метод, який відображає значення `char` у двійковому вигляді. Продемонструйте це, використовуючи кілька різних символів.
- 2.4. Ввести цілі числа n і m вивести ціле число, у якого m -й біт відрізняється від m -го біта числа n , а всі інші біти збігаються з бітами числа n на тих же позиціях. Наприклад, якщо введені 5 і 1, відповіддю буде 7.
- 2.5. Ввести ціле число n (однобайтове число) і вивести число, отримане в результаті циклічного зсуву числа n на один розряд вліво, тобто старший біт посунутий в позицію молодшого, а всі інші біти зсуваються на один розряд вліво. Наприклад, якщо введено 130, відповіддю буде 5.
- 2.6. Визначити, скільки разів зустрічається 11 в двійковому поданні цілого додатнього числа (так, в двійковому поданні 11110111 воно зустрічається 5 разів).
- 2.7. Викреслити i -й біт з двійкового представлення натурального числа (молодші i -го біти залишаються на місці, старші зсуваються на один розряд вправо). Наприклад, якщо введені 11 і 2, відповіддю буде 7.
- 2.8. Ввести два цілих числа то обчислити їх добуток (так щоб він був коректний навіть при максимально великих цілих числах) та частку як дійсне число з максимально можливою кількістю цифр після десяткової крапки.
- 2.9. Введіть довге натуральне число m та натуральне k типу байт та знайдіть k -тий розряд числа m .

- 2.10. Напишіть програму, що вводить два довгих цілих числа та додає та віднімає їх уникаючи проблему переповнення цілого типу.
- 2.11. Напишіть програму, що вводить два цілих числа та виконує арифметичні дії над ними (+, -, *, /, %) таким чином як це повинно виконуватися для беззнакового (натурального) типу.
- 2.12. Ввести натуральне число M . Встановіть її біт з номером j рівним нулевій та виведіть отримане число виведіть отримане число в десятковому та шістнадцятковому вигляді..
- 2.13. Визначить номер першого значущого зліва та справа біта натурального числа M .
- 2.14. Поміняйте місцями перші 8 біт та останні 8 біт натурального числа, виведіть отримане число в десятковому та шістнадцятковому вигляді.. Ввести натуральне число M . Поміняйте місцями біти її двійкового запису з номерами i та j (що теж вводяться) та виведіть отримане число в десятковому та шістнадцятковому вигляді.
- 2.15. Знайдіть кількість значущих (не рівних 0) бітів натурального числа.
- 2.16. Ввести натуральні числа M та N та визначить скільки в них спільних одиниць бітового представлення. Визначить скільки в цих числах взагалі співпадає бітів.
- 2.17. Інвертуйте бітове представлення даного числа та виведіть двійкове представлення та десяткове для цієї інверсії.
- 2.18. Дано три натуральних числа: 32-бітне число, 8-бітний номер біта та булеве значення біта. Необхідно поміняти в першому числі біт з заданим номером на передане значення та вивести отримане після змін біта число в десятковій, шістнадцятковій та двійковій формі.

Приклад 1: 12 1 1

Результат 1: 13 0xD 1101

Приклад 2: 13 1 0

Результат 2: 12 0xC 1100

2.19. Control Bit

На вхід подається ціле число N ($1 \leq N \leq 2^{28} - 1$). Напишіть функцію, що повертає це число в звичайному форматі (big-endian) але кожен байт числа має молодший біт, що визначається іншими 7 бітами як сума по модулю 2 значень кожного з цих 7 бітів (так званий "контрольний біт"). Напишіть також функцію яка по довільному 32-бітному числу повертає відповідне 28-бітне число або повідомлення, що контрольний біт на якомусь байті некоректний.

2.20. Little-Endian, Big-Endian

У сучасному світі будь-які дані в пам'яті комп'ютера і при їх передачі по каналах зв'язку представлені у вигляді послідовності байтів. У різних системах і протоколах байти прийнято упорядковувати по-різному: від старшого до молодшого (big-endian) або від молодшого до старшого (little-endian).

Наприклад:

$$2017_{10} = 11111100001_2$$

Big-endian: $2017_{10} = 00000111\ 11100001_2$ Little-endian: $2017_{10} = 11100001\ 00000111_2$

В блокчейн різницю між кодуванням чисел можна побачити: Bitcoin використовує little-endian, Ethereum використовує big-endian.

Напишіть програму, яка перетворює число до формату з little-endian з урахуванням того для гіпотетичної обчислювальної системи, де використовуються байти складаються тільки з 7 біт.

Формат вхідних даних: На вхід подається ціле число N ($1 \leq N \leq 2^{28} - 1$).

Формат вихідних даних: Необхідно вивести послідовність з 28 символів 0 або 1 (біти) - представлення вхідного числа в форматі little-endian.

Умови

- 3.1. Напишіть програму, яка друкує значення від 1 до 100.
- 3.2. Напишіть програму, яка генерує 25 випадкових значень `int`. Для кожного значення використовуйте оператор `if-else`, щоб класифікувати його як більше, менше або те що дорівнює другому випадково згенерованому значенню.
- 3.3. Напишіть програму `RollDie.java`, яка генерує результат кидання чесного шестигранного кубика (ціле число від 1 до 6).
- 3.4. Напишіть програму, яка бере три цілі аргументи командного рядка `a`, `b` і `c` і надрукуйте кількість різних значень (1, 2 або 3) серед `a`, `b` і `c`.
- 3.5. Напишіть програму, яка бере п'ять цілочисельних аргументів командного рядка та друкує медіану (третю за величиною). Спробуйте обчислити медіану 5 елементів так, щоб після виконання вона ніколи не робила більше 6 загальних порівнянь.
- 3.6. Аліса кидає чесну монету, поки не побачить два послідовних орли. Боб кидає ще одну чесну монету, поки не побачить орла, за яким йде решка. Написати програму для оцінки ймовірності того, що Аліса зробить менше кидків, ніж Боб? Рішення: 39/121.

- 3.7. Перепишіть DayOfWeek.java з вправи 1.2.29 так, щоб він друкував день тижня як неділю, понеділок тощо, замість цілого числа від 0 до 6. Використовуйте оператор switch.
- 3.8. Число-англійська. Напишіть програму для зчитування в цілому числі командного рядка між -999,999,999 та 999,999,999 та виведить англійський еквівалент. Ось вичерпний перелік слів, які має використовувати ваша програма: мінус, нуль, один, два, три, чотири, п'ять, шість, сім, вісім, дев'ять, десять, одинадцять, дванадцять, тринадцять, чотирнадцять, п'ятнадцять, шістнадцять, сімнадцять, вісімнадцять, дев'ятнадцять, двадцять, тридцять, сорок, п'ятдесят, шістдесят, сімдесят, вісімдесят, дев'яносто, сто, тисяча, мільйон.
- 3.9. Оцінка гімнастики визначається колегією з 6 суддів, кожен з яких визначає оцінку від 0,0 до 10,0. Остаточний бал визначається, відкидаючи високі та низькі бали та усереднюючи решту 4. Напишіть програму GymnasticsScorer.java, яка приймає 6 справжніх вхідних даних командного рядка, що представляють 6 балів, і роздруковує їх середнє значення, після того, як викинуло високі та низькі бали.
- 3.10. Скласти програму, яка по колу та прямій встановлює, який випадок має місце: а) дві точки перетину; б) одна точка дотику; в) жодної спільної точки.
- 3.11. Задано два квадрати, сторони яких паралельні координатним осям. З'ясувати, чи перетинаються вони. Якщо так, то знайти координати лівого нижнього та правого верхнього кутів прямокутника, що є їхнім перетином.
- 3.12. Дано два прямокутники, сторони яких паралельні координатним осям. Відомо координати лівого нижнього та правого верхнього кутів кожного з прямокутників. Знайти координати лівого нижнього та правого верхнього кутів мінімального прямокутника, що містить задані прямокутники.

Цикли

- 4.1. Десяткове розкладання раціональних чисел. Враховуючи два цілих числа p і q , десяткове розширення p/q має нескінченно повторюваний цикл. Наприклад, $1/33 = 0,03030303 \dots$. Ми використовуємо позначення $0.(03)$ для позначення того, що 03 повторюється нескінченно довго. Як інший приклад, $8639/70000 = 0,1234(142857)$. Напишіть програму DecimalExpansion.java, яка зчитує два цілих числа командного рядка p і q і друкує десяткове розширення p/q , використовуючи зазначені вище позначення. Підказка: використовуйте правило Флойда.

- 4.2. У п'ятницю, 13 -го. Яка максимальна кількість днів поспіль, у які не буває п'ятниці 13 -го? Підказка: Григоріанський календар повторюється кожні 400 років (146097 днів), тому вам потрібно турбуватися лише про 400 -річний інтервал. Рішення: 426 (наприклад, з 13.08.1999 по 13.10.2000).
- 4.3. 1 січня. Чи більше шансів, що 1 січня випаде на суботу чи неділю? Напишіть програму, щоб визначити, скільки разів кожен з них відбуватиметься з інтервалом 400 років. Рішення: неділя (58 разів) частіше, ніж субота (56 разів).
- 4.4. Напишіть програму NPerLine.java, яка приймає цілий аргумент командного рядка n і друкує цілі числа від 10 до 99 з n цілих чисел на рядок.
- 4.5. Змініть NPerLine.java так, щоб він друкував цілі числа від 1 до 1000 з n цілими числами на рядок. Зробіть цілі числа вирівняними, надрукувавши правильну кількість пробілів перед цілим числом (наприклад, три для 1-9, два для 10-99 і одне для 100-999).
- 4.6. Припустимо, що a, b і c - випадкові числа, рівномірно розподілені між 0 і 1. Яка ймовірність того, що a, b і c утворюють довжину сторони деякого трикутника? Підказка: вони складуть трикутник тоді і тільки тоді, коли сума кожних двох значень більша за третю.
- 4.7. Повторіть попереднє питання, але обчисліть ймовірність того, що отриманий трикутник тупий, враховуючи, що три числа для трикутника. Підказка: три довжини утворюють тупий трикутник тоді і тільки тоді, коли (i) сума кожних двох значень більша за третю та (ii) сума квадратів кожних двох довжин сторін більша або дорівнює квадрату третього.
- 4.8. Напишіть програму AllEqual.java, яка приймає три цілочисельні аргументи командного рядка і друкує рівні, якщо всі три рівні, а в іншому випадку не рівні.
- 4.9. Напишіть програму RollLoadedDie.java, яка друкує результат кидання кубика так, що ймовірність отримання 1, 2, 3, 4 або 5 дорівнює $1/8$, а ймовірність отримання 6 - $3/8$.
- 4.10. Знайдіть всі прості числа з заданого інтервалу.
- 4.11. Знайдіть всі цілі числа Армстронга в даному інтервалі (числа Армстронга - ті що дорівнюють сумі своїх цифр у ступені, що співпадає з розрядністю даного числа)
- 4.12. Число "вампіра" має парну кількість цифр і формується шляхом множення пари чисел, що містять половину числа цифр результату. Цифри беруться з вихідного номера в довільній послідовності. Пари кінцевих нулів не допускаються. Приклади: $1260 = 21 * 60$ $1827 = 21 * 87$ $2187 = 27 * 81$.

Напишіть програму, яка знайде всі 4-значні числа вампірів.

- 4.13. Написати програму для обчислення коренів квадратних рівнянь у дійсних та комплексних числах (тип результату повинен визначати користувач).
- 4.14. Створити клас, що розв'язує систему 2-х лінійних рівнянь
- 4.15. Створити клас, що розв'язує біквадратне рівняння.
- 4.16. Ввести число от 1 до 12 та число від 1 до 31 та число від 2000 до 2100. Перевірити, що це коректна дата та коректний ввід. Вивести цю дату з назвою місяця українською.
- 4.17. Ввести натуральне число (перевірити його натуральність) та підрахувати його факторіал (рекурсивно та циклом)
- 4.18. Ввести послідовність наступним чином: користувачу виводиться напис "a[**]= ", де замість ** стоїть номер числа, що вводиться. Тобто там виводиться написи "a[0]= ", і після знаку рівності користувач вводить число, "a[1]= ", і після знаку рівності користувач вводить число і так далі доки користувач не введе число 0. Після цього потрібно вивести суму введених чисел.
- 4.19. Визначити із скількох від'ємних чисел починається ненульова послідовність цілих чисел, за якою іде 0.
- 4.20. Скласти програми наближеного обчислення суми всіх доданків, абсолютна величина яких не менше $\varepsilon > 0$:
 - а) $y = \ln x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots$;
 - б) $y = \cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots$;
 - в) $y = \sinh(x) = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots$;
 - г) $y = \cosh(x) = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots$;
 - д) $y = e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots$;
 - е) $y = \ln(1+x) = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \dots, (|x| < 1)$;
 - є) $y = \frac{1}{1+x} = 1 - x + x^2 - x^3 + \dots, (|x| < 1)$;
 - ж) $y = \ln \frac{1+x}{1-x} = 2 * \frac{x}{1} + \frac{x^3}{3} + \frac{x^5}{5} + \dots, (|x| < 1)$;
 - з) $y = \frac{1}{(1+x)^2} = 1 - 2 * x + 3 * x^2 - \dots, (|x| < 1)$;
 - і) $y = \frac{1}{(1+x)^3} = 1 - \frac{2*3}{2} x + \frac{3*4}{2} x^2 - \frac{4*5}{2} x^3 + \dots, (|x| < 1)$;
 - й) $y = \frac{1}{1+x^2} = 1 - x^2 + x^4 - x^6 + \dots, (|x| < 1)$;
 - к) $y = \sqrt{1+x} = 1 + \frac{1}{2}x - \frac{1}{2*4}x^2 + \frac{1*3}{2*4*6}x^3 - \dots, (|x| < 1)$;

$$\text{л) } y = \frac{1}{\sqrt{1+x}} = 1 - \frac{1}{2}x + \frac{1*3}{2*4}x^2 - \frac{1*3*5}{2*4*6}x^3 - \dots, (|x| < 1);$$

$$\text{м) } y = \arcsin(x) = x + \frac{1}{2} \frac{x^3}{3!} + \frac{1*3}{2*4} \frac{x^5}{5!} + \dots, (|x| < 1).$$

Вказівка. Суму y обчислювати за допомогою рекурентного співвідношення $S_0 = 0$, $S_k = S_{k-1} + a_k$, $k = 1, 2, \dots$, де a_k – k -тий доданок, для обчислення якого також складається рекурентне співвідношення. В якості умови повторення циклу розглядається умова $|a_k| \geq \varepsilon$.

- 4.21. Обчислити значення експоненти за рядом Тейлора для дійсного числа x ($-20 < x < 20$) та порівняйте зі значенням обчисленим з допомогою бібліотечної функції
- 4.22. Дана непорожня послідовність різних дійсних чисел, серед яких є хоча б одне від'ємне число, за якою йде 0. Визначити величину найбільшого серед від'ємних членів цієї послідовності.
- 4.23. Банк пропонує річну ставку по депозиту A та 15% по вкладу додаються до основної суми депозиту кожен рік. Ви кладете в цей банк D гривень. Скільки років потрібно чекати, щоб сума вкладу зросла до очікуваної суми P ?