

Лабораторна робота № 3.

Тема: Створення та використання класів.

Лабораторне завдання:

1. Одержати індивідуальне завдання.
2. Створити клас відповідно до варіанту та зобразити UML діаграму класу.
3. При створенні класу повинен бути дотриманий принцип інкапсуляції.
4. Створити конструктор за замовчуванням та хоча б два інших конструктори для початкової ініціалізації об'єкта.
5. Створити функції члени згідно з варіантом.
6. Продемонструвати можливості класу.
7. Підготувати звіт про виконання лабораторної роботи.

Індивідуальне завдання.

1. Комплексне число подане парою дійсних чисел (a, b) , де a – дійсна частина, b – уявна частина. Реалізувати клас **Complex** для роботи з комплексними числами. Обов'язково повинні бути реалізовані наступні операції:
 - 1) додавання $\text{add} : (a.b) + (c.d) = (a+c, b+d)$;
 - 2) віднімання $\text{sub} : (a.b) - (c.d) = (a-c, b-d)$;
 - 3) множення $\text{mul} : (a.b) \cdot (c.d) = (ac-bd, ad+bc)$;
 - 4) ділення $\text{div} : (a.b) / (c.d) = (ac+bd, bc-ad) / (c^2+d^2)$;
 - 5) порівняння $\text{equ} : (a,b) = (c,d)$, якщо $(a=c) \ \& \ (b=d)$.
2. Створити клас **Vector3D**, що задається трійкою координат. Обов'язково повинні бути реалізовані такі дії над векторами: додавання і віднімання векторів, скалярний добуток векторів, множення на скаляр, порівняння векторів, обчислення довжини вектора, порівняння довжини векторів.
3. Створити клас **Money** для роботи з грошовими сумами. Число повинне бути подане двома полями: типу `long` для гривень `unsigned char` – для копійок. Дробова частина (копійки) при виводі на екран повинна бути відділеною від цілої частини комою. Реалізувати додавання, віднімання, ділення сум, ділення суми на дробове число, множення на дробове число і операції порівняння.
4. Створити клас **Triangle** для подання трикутника. Поля даних повинні включати кути і сторони. Потрібно реалізувати такі операції: отримання та зміни полів даних, обчислення площі, обчислення периметра, обчислення висот, а також визначення виду трикутника (рівносторонній, рівнобедрений чи прямокутний).
5. Створити клас **Angle** для роботи з кутами на площині, які задаються в градусах та хвилинах. Обов'язково потрібно реалізувати: переведення в радіани, перетворення до діапазону $0^\circ\text{C} - 360^\circ\text{C}$, збільшення та зменшення кута на задану величину, обчислення синуса, порівняння кутів.

6. Рациональний (нескоротний) дріб подається парою цілих чисел (a, b) , де a - чисельник, b - знаменник. Створити клас **Rational** для роботи з раціональними дробами. Обов'язково повинні бути реалізовані операції:

- додавання **add** $(a, b) + (c, d) = (ad + bc, bd)$;
- різниця **sub** $(a, b) - (c, d) = (ad - bc, bd)$;
- множення **mul** $(a, b) \times (c, d) = (ac, bd)$;
- ділення **div** $(a, b) / (c, d) = (ad, bc)$;
- порівняння **equal, greater, less**.

Повинна бути реалізована недоступна функція скорочення дробу **reduce**, що обов'язково викликається при виконанні арифметичних операцій.

7. Створити клас **Date** для роботи з датами в форматі «рік.місяць.день». Дата подається структурою з трьома полями типу unsigned int: для року, місяця і дня. Клас повинен включати не менше трьох функцій ініціалізації: числами, рядком вигляду «рік.місяць.день» (наприклад, «2018.08.31») і датою. Обов'язковими операціями є: обчислення дати через задану кількість днів, віднімання заданої кількості днів з дати, визначення високосного року, присвоєння і одержання окремих частин (рік, місяць, день), порівняння дат (однакові, до, після), обчислення кількості днів між датами.

8. Створити клас **Time** для роботи з часом в форматі «година: хвилина: секунда». Клас повинен включати в себе не менше чотирьох функцій ініціалізації: числами, рядком (наприклад, «23:59:59»), секундами і хвилинами. Обов'язковими операціями є: обчислення різниці між двома моментами часу в секундах, вирахування з часу заданої кількості секунд, порівняння моментів часу, перевід введеного часу в секунди та в хвилини (з округленням до цілої хвилини).

9. Реалізувати клас **FuzzyNumber** для роботи з нечіткими числами, які подаються трійками чисел $(x - e_1, x, x + e_2)$, $e_1, e_2 < x$. Для чисел $A = (A - a_l, A, A + a_r)$ і $B = (B - b_l, B, B + b_r)$ арифметичні операції виконуються за такими формулами:

- додавання: $A + B = (A + B - a_l - b_l, A + B, A + B + a_r + b_r)$;
- віднімання: $A - B = (A - B - a_l - b_l, A - B, A - B + a_r + b_r)$;
- множення $A \cdot B = (A \cdot B - B \cdot a_l - A \cdot b_l + a_l b_l, A \cdot B, A \cdot B + B \cdot a_r + A \cdot b_r + a_r b_r)$;
- обернене число $A = \left(\frac{1}{A + a_r}, \frac{1}{A}, \frac{1}{A - a_l} \right), A > 0$;
- ділення $\frac{A}{B} = \left(\frac{A - a_l}{B + b_r}, \frac{A}{B}, \frac{A + a_r}{B - b_l} \right), B > 0$.

Реалізувати ці операції.

10. Реалізувати клас **Account** для подання банківського рахунку. В класі повинні бути такі поля: прізвище власника, номер рахунку, відсоток нарахувань та сума в гривнях. Відкриття нового рахунку виконується операцією ініціалізації. Необхідно виконати такі операції: змінити прізвище власника рахунку, зняти деяку суму з рахунку, покласти гроші на рахунок, нарахувати проценти, перетворити суму в долари, перетворити суму в євро.

11. Створити клас **Money** для роботи з грошовими сумами. Номінали купюр: 1, 2, 5, 10, 20, 50, 100, 200, 500 грн. Копійки 0.01 (1 копійка), 0.05 (5 копійок), 0.1 (10 копійок), 0.25 (25 копійок), 0.5 (50 копійок). Сума повинна бути подана полями-номіналами, значенням яких є кількість відповідних купюр. Реалізувати додавання сум, віднімання сум, ділення сум, ділення суми на дробове число, множення на дробове число і операцію порівняння. Дробова

частина (копійки) при виведенні на екран повинна відділятися від цілої частини комою.

12. Реалізувати клас **Bankomat**, який моделює роботу банкомата. В класі повинні міститися поля для ідентифікаційного номера банкомата, інформація про наявну в банкоматі поточну суму грошей, мінімальну і максимальну суму, яку дозволено зняти клієнту за день. Сума грошей подається полями-номіналами (див. завдання 3). Реалізувати метод ініціалізації банкомата, метод завантаження купюр в банкомат і метод зняття певної суми грошей. Метод зняття грошей повинен виконувати перевірку на коректність суми, яка знімається (ця сума не повинна перевищувати максимальне значення).

13. Створити клас **Fraction** для роботи з дробовими числами. Число подається двома полями: ціла частина – довге ціле зі знаком, дробова частина – без знакове коротке ціле. Реалізувати арифметичні операції додавання, віднімання, множення і операцію порівняння.

14. Створити клас **Goods** (товар). В класі повинні бути подані поля: найменування товару, дата оформлення, ціна товару, кількість одиниць товару, номер накладної, за якою товар поступив на склад. Реалізувати методи зміни ціни товару, зміни кількості товару (збільшення, зменшення), обчислення вартості товару.

15. Створити клас **Payment** (зарплата). В класі повинні бути подані поля: прізвище-ім'я-по-батькові, оклад, рік влаштування на роботу, відсоток надбавки, податок, кількість відпрацьованих днів за місяць, нарахована і відрахована суми. Реалізувати методи обчислення нарахованої суми, обчислення відрахованої суми, обчислення суми до виплати, обчислення стажу. Стаж обчислюється як кількість повних років, які минули з моменту влаштування на роботу до поточного року. Нарахування – це сума, нарахована за відпрацьовані дні та надбавки, тобто відсоток від першої суми. Відрахування – це відрахування в пенсійний фонд (1% від нарахованої суми) і податок, який становить 18% від нарахованої суми без відрахувань в пенсійний фонд.