

8. ООП (об'єктно-орієнтоване програмування)

1) Описати клас Поліном та реалізувати методи: введення поліному, виведення поліному, обчислення значення поліному у точці x , взяття похідної поліному, суми, різниці та добутку поліномів.

Використати цей клас для розв'язання задачі: ввести 2 поліноми P_1 , P_2 та рядок, який містить вираз, що залежить від 2 поліномів. Наприклад,

$$P_1 + P_2 * P_1 - P_2$$

Обчислити поліном, який буде значенням цього виразу.

Вказівка: поліном представити у вигляді масиву змінної довжини.

- 1) Опишіть класи для таких геометричних фігур та реалізуйте зазначені методи:
 - a) Клас Трапеція. У цьому класі реалізуйте операції знаходження периметра і площі;
 - b) Клас Паралелограм. У цьому класі реалізуйте операції знаходження периметра і площі.
 - c) Клас Круг. Реалізуйте методи відшукування площі круга, довжини кола, цього круга.
 - d) Клас Піраміда. Реалізуйте методи пошуку площі бічної поверхні і об'єму;
 - e) Клас П'ятикутник, що містить список вершин. Реалізуйте метод перевірки чи є цей п'ятикутник опуклим.
 - f) Клас Багатокутник. Реалізуйте метод перевірки чи є цей багатокутник опуклим.

Дано список фігур вищенаведених класів. Серед фігур, що належать до перших трьох класів знайдіть фігуру, що має найбільшу площу та периметр (довжину кола). Також знайдіть всі опуклі багатокутники

3) Описати клас **Прямокутник**. Сторони прямокутника паралельні осям координат. Для прямокутника задані лівий верхній кут та довжини сторін. Описати методи отримання довжини кожної з сторін, площі прямокутника, периметру, метод знаходження перетину двох прямокутників. Методи переміщення прямокутника. Скласти програму створення заданої кількості прямокутників та знаходження їх спільного перетину.

4) Описати клас **Трикутник**. Основа трикутника паралельна осі x координат. Для трикутника задані лівий нижній кут (координати) та довжини сторін. Описати методи отримання довжини кожної зі сторін. Описати методи отримання довжини кожної з сторін, площі прямокутника, периметру, метод знаходження перетину двох прямокутників. Методи переміщення

прямокутника. Скласти програму створення заданої кількості прямокутників та знаходження їх спільного перетину.

5) Описати клас **Трикутник**. Основа трикутника паралельна осі x координат. Для трикутника задані лівий нижній кут (координати) та довжини сторін. Описати методи отримання довжини кожної зі сторін. Описати методи отримання довжини кожної з сторін, площі, периметру, метод знаходження перетину двох трикутників. Методи переміщення. Скласти програму створення заданої кількості трикутників та знаходження їх спільного перетину.

6) Описати клас **Еліпс**. Для нього задані фокуси та радіуси. Описати методи отримання геометричних характеристик. Описати методи отримання довжини радіусів, площі, периметру, метод знаходження площі перетину двох еліпсів. Методи переміщення та повороту. Скласти програму створення заданої кількості еліпсів та знаходження їх спільного перетину.

7) Опишіть класи

1. **Гість**, що містить всю необхідну інформацію про жильця деякого готелю: ім'я, період проживання тощо.
2. **Кімната**, що містить інформацію про кімнату готелю у тому числі вартість проживання за добу.
3. **Готель**, що містить список кімнат цього готелю, інформацію про те ким і коли вони зайняті, а також методи на кшталт тощо.

Використовуючи вищенаведені класи розв'язати задачі:

- а) Вивести відомість про кількість вільних кімнат у готелі;
- б) Пошуку вільної кімнати у зазначений період;
- в) Поселити жильця на вказаний термін;
- г) Вартості проживання жильця у зазначений період;
- д) Прибутку, який отримає готель за вказаний період;
- е) Пошуку гостя у готелі (у заданий період);

8)Зробити клас СЛОВО, який має члени ПРИСТАВКА, ПРИСТАВКА2, КОРІНЬ, СУФІКС, ЗАКІНЧЕННЯ (гетери сеттери).

Наслідники цього класу: ГЛАГОЛ, ЧИСЛІВНИК, ЗАЙМЕННИК, ІМЕННИК, ПРИЙМЕННИК, ПРИКМЕТНИК, АРТИКЛЬ.

Для них методи Род, Число, Лице, Відмінок
Функції

Клас РЕЧЕННЯ: члени ПОДЛ, СКАЗ, ДОПОВНЕННЯ(ДОПЧАСУ, ДОПМІСЦЯ, ДОПОБСТАВИНИ)

9) Опишіть клас Фігура, що інкапсулює основні геометричні характеристики та методи. Для фігури визначено методи:

1. `calculateVolume()` – віртуальний метод, що обчислює міру фігури (для плоскої фігури – площу, для об'ємної – відповідно об'єм).
2. `getVolume()` – що повертає міру фігури.

Від класу Фігура наслідуються такі класи

- Трикутник
- Прямокутник
- Трапеція
- Паралелограм
- Круг
- Куля
- Трикутна Піраміда (який наслідується від класу Трикутник)
- Чотирикутна піраміда (який наслідується від класу Прямокутник)
- Паралелепіпед (який наслідується від класу Прямокутник)

Нехай дано список фігур. Серед заданих фігур, знайдіть фігуру, що має найбільшу міра якої є найбільшою

- 1) Опишіть клас `Pet` – домашня тварина, що має метод `to_feed(feed, count)` – годувати (`feed` – тип корму, `count` – кількість).

Клас `Pet` має віртуальні методи

1. `to_sniff()` («нюхати» – визначає, чи може їсти тварина заданий тип корму),
2. `to_ask()` («просити» – метод повертає `True`, якщо тип корму не підходить або тварина ще хоче їсти і виводить на екран прохання «тваринною мовою», наприклад, «М'яв...» для кота),
3. `to_eat()` (їсти, якщо тип корму підходить).

Клас `Pet` має нащадки – `Cat`, `Dog`, `Parrot` (папуга), у яких перевизначено вищезгадані віртуальні методи.

Задано список тварин та список кормів (тип та загальна вага). Пропонуючи по черзі кожній тварині порцію їжі, потрібно нагодувати всіх тварин. Якщо корму не вистачить – вивести відповідне повідомлення.

11) Опишіть клас `Car`, що має метод `go(distance)`, який змінює пройдений кілометраж автомобілем та залишок пального. Метод `go(...)` залежить від віртуального методу `fuelPerKm()`, який визначає скільки потрібно пального автомобілю для проїзду одного кілометра. Нехай `Personal` (легковий автомобіль) і `Truck` (вантажівка) – класи, що наслідують клас `Car` і перевизначають метод `fuelPerKm()`. При цьому потрібно врахувати, що цей метод залежить від кількості пасажирів (+10% на кожного пасажирів) для авто класу `Personal` або ваги вантажу для `Truck` (+25% на кожну тону вантажу). Визначити чи зможе задане авто проїхати задану відстань.

12) Задано клас `Flower`, що має нащадками конкретні класи квітів (напр., тюльпан, троянд, тощо). Ви зайшли у квітковий магазин у якому продаються

різні типи квітів. Необхідно зібрати букет з квітів (букет може містити квітки одного класу) та визначити:

1. Його вартість.
2. Скільки часу зможе тішити букет очі (до моменту поки не зів'яне перша квітка).
3. Колір, що домінує у цьому букеті.
4. Чи припустимий цей букет за інтенсивністю запаху.

Для наступних задач будемо вважати, що клас Person описано таким чином:

```
class Person{ //Клас Особа
```

```
    string name; //прізвище  
    unsigned byear; //рік народження
```

```
public:
```

```
    int input(){ //ввести особу  
        cin>>name;  
        cin>>byear;  
    }
```

```
    void print(){// #вивести особу  
        cout<<name<<" "<<byear<<endl;  
    }
```

13) Описати клас Знайомий на базі класу Person.

У цьому класі повинно бути як мінімум одне додаткове поле «номер телефону» а також методи введення та виведення інформації про знайомого. Використати цей клас для побудови телефонного довідника.

Передбачити дії: створення довідника, додавання запису про знайомого, пошуку номера телефону за прізвищем та заміни номера телефону.

Телефонний довідник зберігати у файлі та читати з файлу.

Вказівка: телефонний довідник представити у вигляді списку знайомих. Дані зчитувати з (текстового) файлу.

1) Описати клас Пасажир на базі класу Person.

У класі Пасажир повинно бути як мінімум два додаткових поля: «місто відправлення» та «місто прибуття», а також методи введення та виведення інформації про пасажирів та розрахунку плати за білет. Використати цей клас для розрахунку плати за білети усіх пасажирів. Вважати що маршрути зберігаються у списку кортежів (місто1, місто2, відстань), а також те, що плата за білет пропорційна відстані та відома плата за 1 км відстані.

Вказівка: інформацію про пасажирів представити у вигляді списку. Дані зчитувати з (текстового) файлу

2) Описати клас Студент на базі класу Person.

У класі Студент повинна бути інформація про оцінки отримані ним протягом сесії (за 5-ти бальною та 100 бальною шкалами).

Скласти програму для обчислення нарахованої студентам стипендії в залежності від результатів сесії:

- За старим підходом нарахування стипендії (середній бал за всі іспити має бути не меншим ніж 4 за 5-ти бальною шкалою).
- З новим підходом нарахування стипендії (стипендію отримують 40% від загального числа студентів, які є найкращими по рейтингу)

Вказівка: інформацію про студентів представити у вигляді списку. Дані зчитувати з (текстового) файлу.