1. Які принципи ООП ви знаєте? Де вони проявляються?

-інкапсуляція Encapsulation - властивість пов"язати дані та методи роботи всереди класу (в клас ми вкладаємо змінні, що характеризують обєкт і методи работи з цими змінними)

- змінні члени класу треба приховати to private

- далі в паблік пишемо методи роботи з змінними

-наслідування inheritance - роблять базовий клас і на основі його роблять дочірниій клас

-поліморфізм - економія понять. Коли у багатьох класах метод називається однаково а реалізацію має свою.

2. Що таке інкапсуляція? Чому важливо її дотримуватися та як зробити?

3. Що таке наслідування? Які його завдання?

4. Що таке поліморфізм?

class Animal

{

protected:

string name;

int age;

public:

Animal() : name("none"), age(0) {};

Animal(string n, int a) :name(n), age(a) {};

void setName(string n) {

this->name = n;

}

string getName() { return name; };

void show() {

cout << "Name: " << name << "\nAge" << age << endl;

}

virtual void Run() {

cout << "I am running" << endl;

}

};

class Rabit:public Animal

{

int speed;

public:

Rabit() :Animal(), speed(0) {};

Rabit(string n, int a, int s) :Animal(n, a), speed(0) {};

void show() {

Animal::show;

cout << "Speed: " << speed << endl;

}

void Run() override

{

cout << "I am running faster" << endl;

}

};=============================================

5. В чому особливість static метода або поля в класі?

Статисчна змінна - створюється один раз і стираєтся після роботи програми (або після закриття {}

Якщо це в классі то використовується для рахування кількості обєктів цього класу і треба перед мейном зробити цю змінну

Статичний метод - в класі перед методом ставится static, і він є доступний для виклику без звернення до конкретного обєкту.

6. В чому різниця між структурою та класом?

Коли є не складна (імя фамілія). Створюєм конструктор дефолтний, параметризований і функція принт. Якщо щось кладніше то класс

7. Що таке const метод?

Методи що не міняють this

8. Що таке this і яка в нього особливість в const методах?

Вказівник на обєкт классу

9. В якому типу метода ми не маємо доступу до вказівника this?

В статичних методах

10. Яким чином можна перевантажити постфіксну та префіксну форму інкремента?

Фіктивним параметром. В префіксній формі просто оператор ++() - пусті дужки. В постфіксній в (int a) - фіктивний параметр.

11. Як визначається блок catch, який повинен спрацювати при генерації exception?

По типу даних

12. Яким чином можна обробити будь-яким тип виключення?

оператор три крапочки

13. Яка відмінність в роботі оператора throw та return?

Throw кидає тільки помилки, а return значення. Але обидві закінчують метод.

14. В чому різниця між ініціалізацією та присвоєнням?

Ініціалізація - коли створили змінну без значення

Ініціалізація з присвоєнням - створили змінну одразу вказавши значення

Присвоєння - через = зберігли значення

15. Яким чином можна ініціалізувати параметри класа через конструткор?

Конструктори можуть бути розгорнутими або через ліст ініціалізації

17. В чому різниця між конструктором копіювання та переміщення? Коли вони необхідні?

Конструктор копіювання робить точну копію обєкта, а конструктор переміщення не робить нову копію, він дає нову адресу на ориганал

18. Що таке explicit?

Заборонити неявне перетворення від одного типу даних до іншого, пишеться в конструкторі з одним параметром

19. Що означають ключові слова delete та default для опису методів класа?

Delete - заборонити дефолтний конструктор

Default - навпаки, вказуємо щоб був дефолтний конструктор (бо коли є хоч один параметризований, то дефолтного не буде)

Ставиться в конструкторах, деструкторі, оператор =

20. В чому особливість дружніх функцій? Під яким специфікатором доступу їх потрібно описувати?

Дає доступ функції ззовні доступ до приватних змінних класу

21. В чому різниця між агрегацією та композицією? Як їх можна реалізувати?

Композиція в програмуванні- це коли один клас складається з інших класів, і вони є невідємною його частиною. (Наприклад, є принтер. Окремо є класс сканер і класс картридж. В классі принтер є сканер, і массив картриджей)

Агрегація - це взяємодія між классами коли звязок не обовязковий, може бути, може не бути (принтер без файлу для друку норм себе почуває, друк може бути а може і не бути) - робим пустий вказівничок і в разі чого додаємо

22. Яким чином можна отримати явний доступ до елементів глобальної області видимості?

::A - оператор дві крапочки

24. Для чого буває потрібно перевантажувати оператор за допомогою функції?

Щоб навчити компілятор працювати з власно створенними типами даних

Наслідування та поліморфізм

26. Які види наслідування ви знаєте?

Одинарне

Множинне

27. Які проблеми можуть виникнути при використанні множинного наслідування?

Є батьківський, два наслідники, і клас наслідних цих двох. Може приймати змінні і методи з самого батьківського классу два рази

28. Що таке абстрактний клас?

Обстрактний клас - де є чисто віртуальний метод

29. Для означають специфікатори override та virtual?

Virtual перед назвою метода дозволяють перевизначити ці функції (зробити свій функціонал)

override - ставиться після перевизначенної функції щоб перевірити чи є в батьківському классі

30. В чому особливість віртуальних функцій?

просто virtual дозволяє перевизначення

Чисто віртуальна функція не має ніякої реалізації

31. Що таке чисто віртуальна функція? Коли її доцільно використовувати?

Чисто віртуальна функція не має ніякої реалізації. Функція є, а як її реалізувати поки не ясно

32. Що таке перевизначення функції? Що для цього потрібно?

В батьківському одна реалізація в насліднику інша. Virtual override

33. В якому порядку викликаються конструктори та деструткори при створенні класа, який успадкував інші класи?

Батьківський --> дочірня

34. Які специфікатори наслідування ви знаєте? В чому різниця між ними?

private - немає доусупу

pritected - доступ є у дочірньому класу

public - вільний доступ

35. В чому особливість віртуального деструктора? Для чого він потрібний?

Якщо в батьківському класі є динамічний масив і в дочірньому - щоб прописати правильне видалення дин памяті

36. В чому особливість віртуального конструктора?

немає такого

Динамічні структури даних

38. В чому різниця між масивом та списком? На що потрібно орієнтуватися при виборі?

масив - комірки послыдовно друг за другом

шукати і сортувати

ліст - комірки рандомно і знають адерсу попередньої і наступної комірок

додавати і видаляти краще

39. Що таке дерево сортування? Для чого можна використовувати?

Бінарне дерево

40. В чому різниця між map та multimap?

map - асоціативний контейнер (на основі бінарного дерева, відповідно дані посортовані)

Зберігає як слованик на пайтоні.

Йде ключ і значення

!!!!!!!Ключ має бути унікальний!!!!!!!!

!!!!в мапі ключі зміняти не можна!!!

41. Якою властивістю повинен володіти об'єкт, щоб його можна було додати в асоціативні контейнери в якості ключа?

Унікальний

42. Які види черг ви знаєте? В чому відмінність?

Звичайна, кругова, з пріоритетним включенням і з пріоритетним виключенням

43. Що означає характеристика size та capacity в контейнерах?

сайз - кількість наявних елементів, капасіті - к-сть зарезервованих комірок

44. Яка структура контейнера vector?

Динамічний масив, але сам управляє памятю, додає видаляє

46. Що таке ітератор? Які види ітераторів існують? В чому їх відмінність?

Ітератор - це вказівник на початок або на кінець