Контрольные вопросы 11.11.23 Анисимов Антон

1.

Преимущества:

- Есть иерархия классов, явный интерфейс

- Выполнение в runtime (т.е компиляция будет быстрее, чем в случае статического)

- Работа с объектами разных типов

Недостатки:

- Оверхеды, связанные с использованием таблиц виртуальных функций

- При работе со множеством больших объектов скорость выполнения существенно снизится

2.

Преимущества:

- Нет оверхедов, связанных с vtables

- Полиморфизм времени компиляции, т.о в runtime выполнение программы будет быстрее

Недостатки:

- Иногда после инстанцирования шаблонные классы или функции имеют разные типы, т.о в один контейнер их не всегда можно положить, т.е пакетно обрабатывать будет проблематичнее

- Уходит больше времени на компиляцию

- Отсутствие иерархии и явных интерфейсов

- Раздутие кода

3.

- Полиморфизм – статический, что привносит все его преимущества

- Возможность определить набор обязательных функций при использовании статического полиморфизма, т.о сделать интерфейс явным, возможность сделать код иерархичным – т.о часть недостатков стат.полиморфизма исправлена

- Обобщенное программирование

4.

Можно использовать mixin - прием проектирования, когда класс (интерфейс, модуль) реализовывает некоторый функционал, который потом можно привнести в другой класс. Пример простейшего миксина:

#include <iostream>

class Base {

public:

void basen() {

std::cout << "Base \n";

}

};

class Add {

public:

void add () {

std::cout << "Add\n";

}

};

class Mixed : public Base, public Add {

public:

void mix {

std::cout << "mix\n";

}

};

5.

- Если нужно реализовать контейнер, работающий с произвольным количеством аргументов разных типов

- Если нужно реализовать шаблоны классов или функций, у которых произвольное число аргументов разных типов.