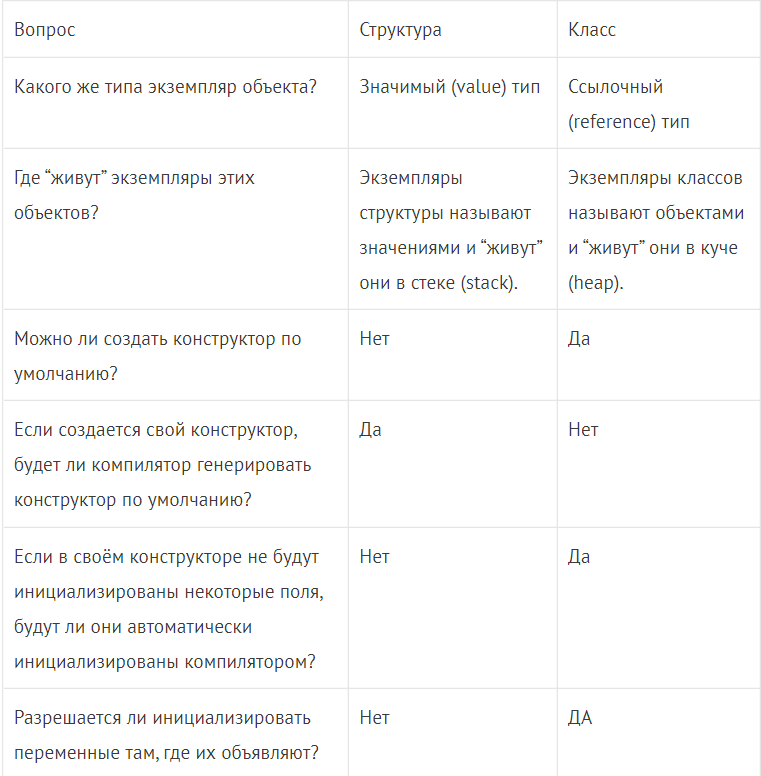
ВОПРОСЫ К ЛАБОРАТОРНОЙ РАБОТЕ №5:

1. Чем класс отличается от структуры?

СТРУКТУРА:

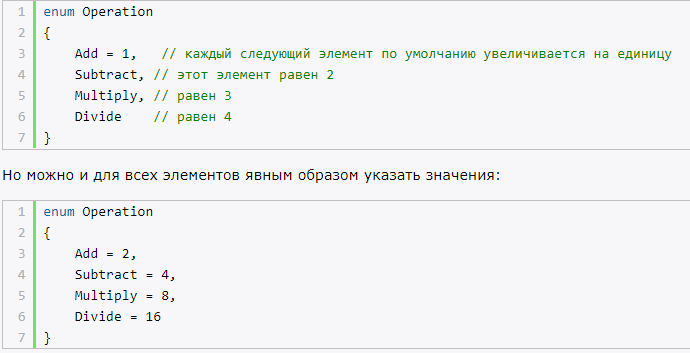
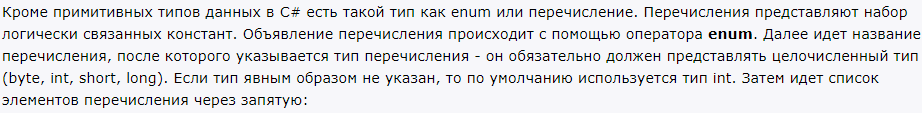
* нельзя инициализировать поля при объявлении
* нет авто- инициализаций полей компилятором
* нет наследования



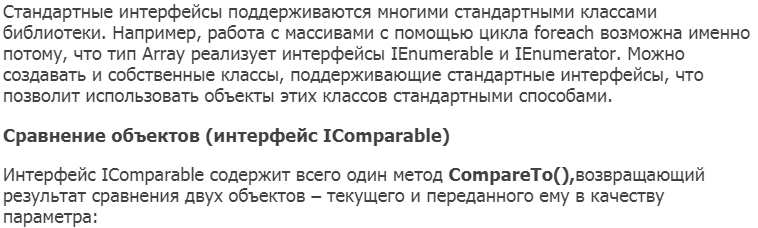
2) Что может и чего не может быть в структуре?

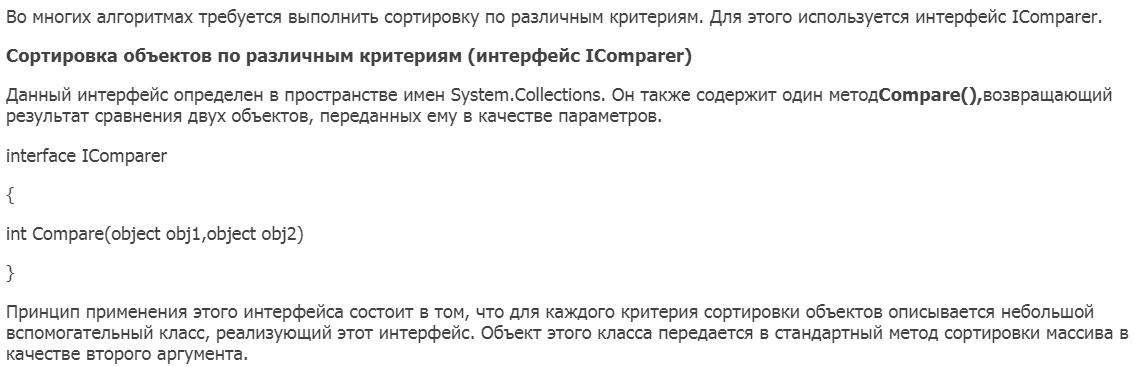
* может иметь конструктор с параметрами
* нельзя конструктор по умолчанию (без парам.) ибо авто-
* Нельзя инициализировать поля структуры при объявлении
* нет автоматической инициализации полей компилятором
* структуры не поддерживают наследование
* Могут реализовывать интерфейсы
* нельзя объявить деструктор

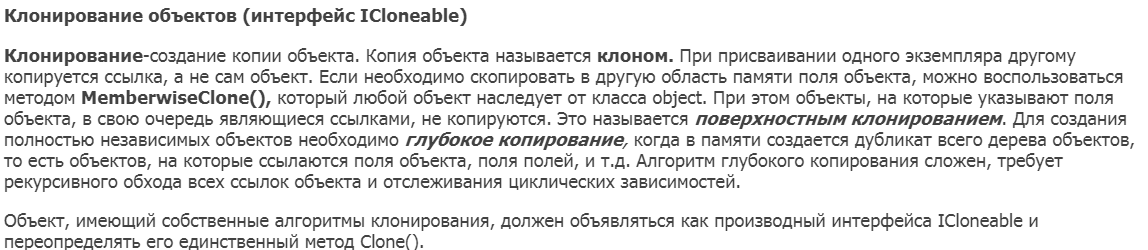
3) Что такое перечисление?



4) Перечислите и поясните стандартные интерфейсы .Net?



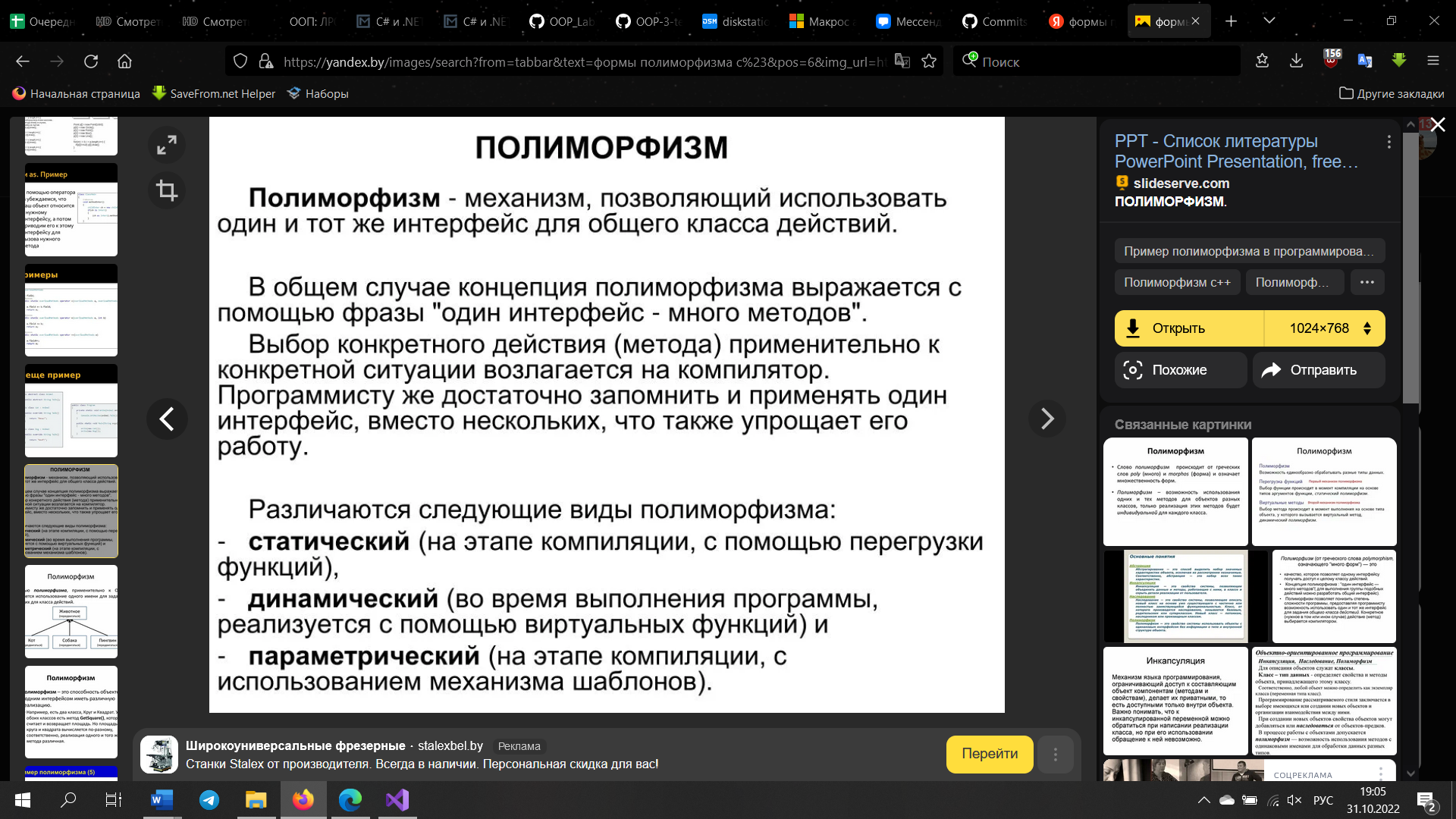


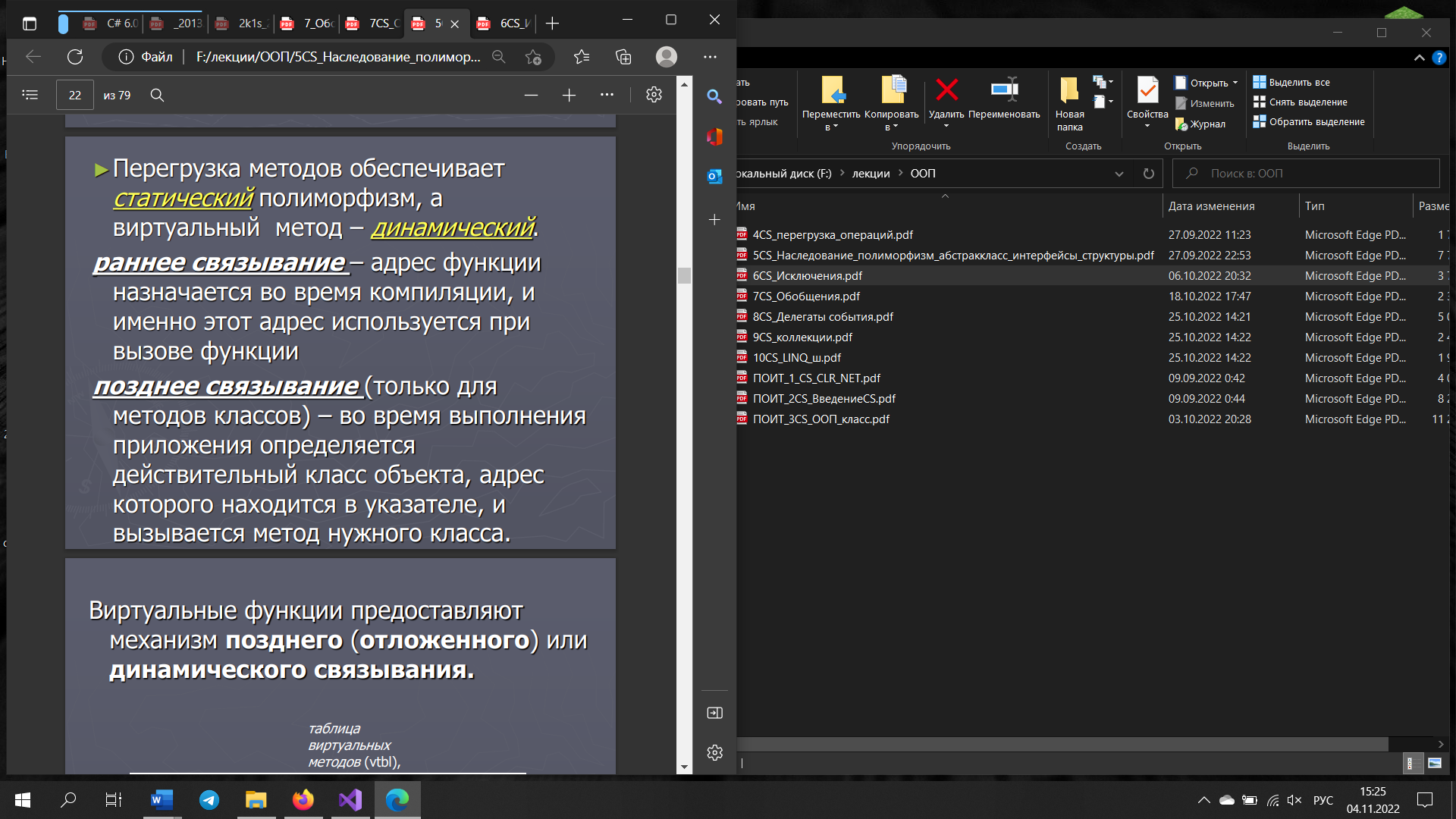


|  |  |  |  |
| --- | --- | --- | --- |
| Интерфейс | Методы | Назначение | Применение |
| **IClonable** | object Clone() | клонирование объектов (поверх. и глуб.) |  |
| **IEnumerable** | (IEnumerator) GetEnumerator(); | перебор эл-тов необобщенной коллекции | основной для большинства коллекций |
| **IEnumerator** | Current bool MoveNext() void Reset() | перебор по необобщенной коллекции | можем перебирать объекты в цикле foreach |
| **IComparable** | int CompareTo (object obj) | для срав. объектов для выяснение порядка |  |
| **IComparer** | int Compare (object o1, object o2) | сравн. объектов | будут иметь больший приоритет |
| **IDisposable** | void Dispose() | механизм для освобождения |  |
| **IEquatable<T>** | bool Equas (T other) | срав. объектов на рав-во |  |
| **IStructuralEqu-atable** | bool Equals (object, IEqualityComparer ); int GetHashCode(…) | проверка на равенство по значению |  |

7) Что такое полиморфизм? Перечислите его формы. Приведите примеры.

**Полиморфи́зм** — возможность объектов с одинаковой спецификацией иметь различную реализацию, переопределение членов баз. класса.





8) Зачем в классе определяют виртуальные методы?

Чтобы их можно было **переопределить**.



**override** – изменение абстрактной или виртуальной реализации  
**virtual** – разрешает переопределение (↑)

9) Как сделать запрет переопределения методов?

С помощью ключевого слова **sealed**.

public sealed override int Sum()