//ca

ООП

1 поколение языков: были

Бейсик

2 поколение:

Код может работать с общими данными и данными интегрированы , процедуры передачи данных функции

3 поколение: все куски кода имеют собственные данные и есть понятие модульность (паскаль, си)

4 поколение c++, понятие «объект» и есть поколение 4+(js, c#)

ООП – использование пользовательских типов данных, классов, которые объединяют данные и механизмы работы с ними

Парадигмы программировании:

1. Наследование – механизм, который позволяет уточнить набор данных и механизмов работы с ними в классах наследников(добавить новое)

2. Инкапсуляция – механизм, которые позволяет скрыть данные и способы работ с ними от пользователя(чтобы упростить восприятие кода, безопасность, чтобы пользователь не смог использовать во вред)

//ca

**Полиморфизм** — методы работы с данными могут иметь разное содержание в зависимости от точки нахождения.

Класс (в С++) тип данных, разработанный пользователем и содержащий (может содержать) данные описания данных и способы работы с ними (методы). Класс — не более чем описание.

Объект — физическая реализация “переменной” соответствующего типа класса. Кусок памяти, в котором находятся данные, но не методы. Но объект может вызвать методы класса для работы со своими данными.

В С++ объекты находятся по одному адресу. Увеличился объем памяти не равно увеличился сам объект. Он имеет фиксированный размер, который создается при реализации класса.

Полиморфизм: есть один механизм, но его реализация будет различной для разных классов.

Язык С++ появился для машин mainframe’ов. Нужно было сделать строгую типизацию.