**Конструктор копирования**

Конструктор копирования –специальный конструктор, который получает в качестве параметра константную ссылку на объект этого же типа:

**class**Person {

…

**public**:

Person(**const Person&**);

*// конструктор копирования*

…

};

**Конструктор копирования вызывается:**

**1.при создании нового объекта с инициализацией существующим объектом:**

Person n1(”Serge Ivanov”);

Person n2(n1);

*// запуск конструктора копирования*

Person n3 = n1;

*// также запуск конструктора копирования*

**2.при передаче в функцию объекта по значению:**

**void** my\_func(Person t) {

…

}…

Person n1(”Serge Ivanov”);

…

My\_func(n1);

*// формальный параметр tсоздается из объекта n1*

*// конструктором копирования*

**3.при выходе из функции, возвращающей объект:**

Person my\_func() {

**char** s[100];

*//формирование имени*

Person t(s);

…

**return**t;

*//объект t будет уничтожен после выхода из*

*// функции, но перед этим на его основе*

*// конструктором копированиябудет сформирован*

*// результат*

}

…

cout<< my\_func().GetName();

**Когда нужно писать конструктор копирования?**Если конструктор копирования не написан, работает т.н. ***стандартный конструктор копирования***

Стандартный конструктор копирует значения всех полей источника в создаваемый объект. Если этого достаточно, писать собственный конструктор копирования не нужно.