

# Язык C++

Мещерин Илья

## Лекция 6

### Наследование

#### 5.1) Объявление

*Derived* наследник *Base*

```
class Derived : public Base{  
  
};
```

- *public* - публичное наследование (все функции из вне класса знают о том, что класс является наследником и могут вызывать методы родителя, но, конечно, только те методы, которые у родителя являются публичными)
- *private* - приватное наследование (сам класс знает от кого наследуется, но никто из вне не знает)
- *protected* - protected наследование (сам класс знает от кого наследуется, а так же его потомки то же об этом знают)

#### 5.2) Поиск имен при наследовании

##### 5.2a) Соккрытие имен наследником

```
class Base{  
public:  
    void f();  
};  
  
class Derived : public Base{  
public:  
    void f();  
};  
  
int main(){  
    Derived d;  
    d.f();  
}
```

В данном случае вызовется версия функции *f()*, описанная в классе *Derived*, что логично

```
class Base{  
public:  
    void f();  
};  
  
class Derived : public Base{  
public:
```

```
    void f(int x);  
};  
  
int main(){  
    Derived d;  
    d.f();  
}
```

Ошибка компиляции. Правило поиска имен - имя ищется сначала в классе *Derived* и все функции с таким именем участвуют в перегрузке и только они.