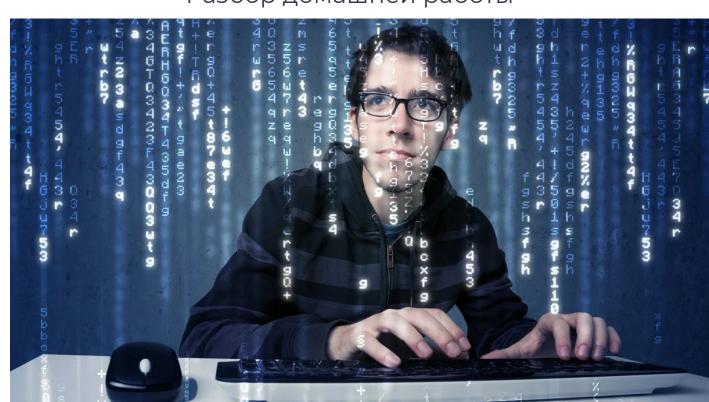




Разбор домашней работы



Методы



Теория







Методы

Метод в объектно-ориентированном программировании - функция, принадлежащая какому-то классу.

Метод имеет доступ к данным объекта, к которому будет применяться.

6 ====

Зачем могут быть нужны методы?

Встречались ли вам методы ранее?

```
class Point {
public:
    double x, y;
    void print() {
        cout << x << " " << y << endl;
    void move(double dx, double dy) {
        x += dx;
        y += dy;
```



```
int main() {
    Point a;
    a.x = 1;
    a.y = 3;
    a.move(5, -4);
    a.print();
    return 0;
}
```

Что будет выведено?

```
int main() {
    Point a;
    a.x = 1;
    a.y = 3;
    a.move(5, -4);
    a.print();
    return 0;
}
```

Что будет выведено?

6 -1

11 ====

Какие ещё методы можно ? было бы добавить?



Практика







Теория





Как передать объект класса в метод?



Передача объектов в функции

Передача объектов классов в методы ничем не отличается от передачи любой другой переменной.

Если необходимо изменить объект внутри метода, его необходимо передать по ссылке.

```
class Vector {
public:
    double x, y;
    void makeVector(Point a, Point b) {
        x = b.x - a.x;
        y = b.y - a.y;
```

```
class Point {
public:
    double x, y;
    void print() {
        cout << x << " " << y << endl;
    void move(Point d) {
        x = d.x;
        y = d.y;
```



Может ли метод вернуть ? какое-либо значение?

```
class Point {
public:
    double x, y;
    //Методы, описанные ранее
    double distance(Point p) {
        return sqrt(pow(x - p.x, 2) + pow(y - p.y, 2));
cout << a.distance(b);</pre>
```



Практика





Итоги урока

- 1) Что такое метод
- 2) Как создать метод
- 3) Отличие метода от функции
- 4) Передача объектов в методы