

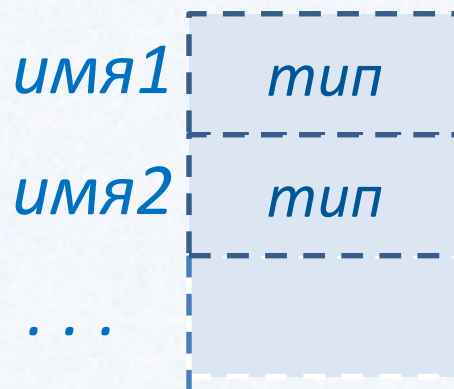
## ***C4. Структуры***

# Определение структуры

```
struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    . . .  
};
```

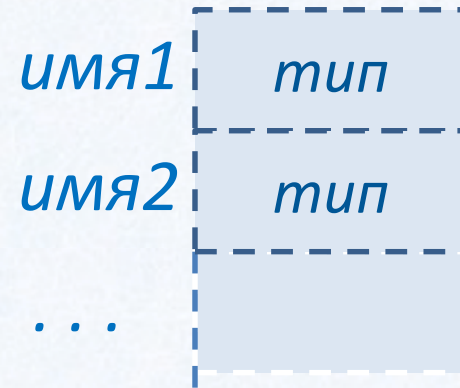
# Определение структуры

```
struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    ...  
};
```



# Определение структуры

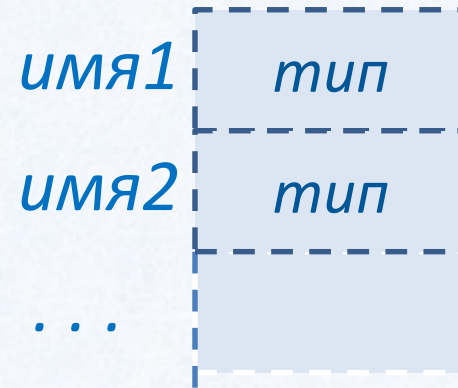
```
struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    ...  
};
```



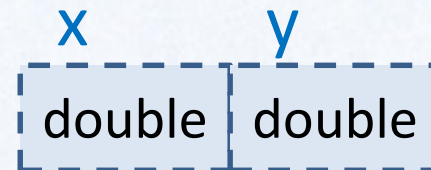
```
struct Point {  
    double x, y;  
};
```

# Определение структуры

```
struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    ...  
};
```



```
struct Point {  
    double x, y;  
};
```



# Определение структуры

```
typedef struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    . . .  
} новое_имя_типа;
```

# Определение структуры

```
typedef struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    . . .  
} новое_имя_типа;
```

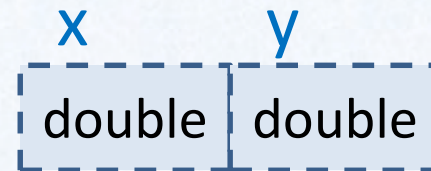
```
typedef struct Point {  
    double x, y;  
} Point;
```



# Определение структуры

```
typedef struct имя_структуры {  
    тип имя1, ... ;  
    тип имя2, ... ;  
    . . .  
} новое_имя_типа;
```

```
typedef struct Point {  
    double x, y;  
} Point;
```





## С4.3 Определение переменных

struct *имя\_структуры* *имя\_переменной*

## С4.3 Определение переменных

```
struct имя_структуры имя_переменной  
    [= {значение_1, значение_2, ... }];
```

## С4.3 Определение переменных

```
struct имя_структуры имя_переменной  
    [= {значение_1, значение_2, ... }];
```

```
struct Point p1 = {1.25, -3.8}, p2;
```

## C4.3 Определение переменных

struct *имя\_структуры имя\_переменной*  
[ *{значение\_1, значение\_2, ...}* ] ;

struct Point p1 = {1.25, -3.8}, p2;

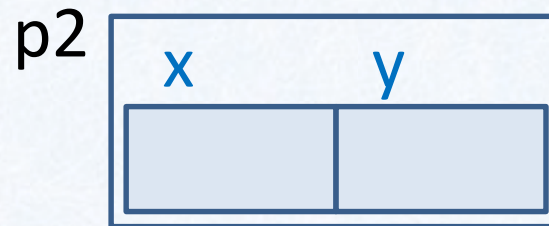
p1

x	y
1.25	-3.8

## C4.3 Определение переменных

struct *имя\_структуры* *имя\_переменной*  
[ *{значение\_1, значение\_2, ...}* ] ;

struct Point p1 = {1.25, -3.8}, p2;



## С4.4 Определение переменных

Если использовался typedef:

## С4.4 Определение переменных

Если использовался typedef:

*имя\_структуры имя\_переменной*  
*[ = {значение\_1, значение\_2, ... } ] ;*



## С4.4 Определение переменных

Если использовался typedef:

*имя\_структуры имя\_переменной*  
*[ = {значение\_1, значение\_2, ... } ] ;*

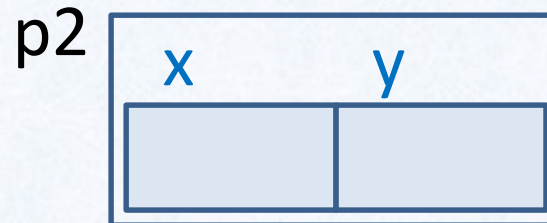
Point p1 = {1.25, -3.8}, p2;

## С4.4 Определение переменных

Если использовался typedef:

*имя\_структуры имя\_переменной*  
*[ = {значение\_1, значение\_2, ... } ] ;*

Point p1 = {1.25, -3.8}, p2;



# Определение массива

```
struct имя_структуры имя_массива[количество]
```

# Определение массива

```
struct имя_структуры имя_массива[количество]  
[  
= {  
  
  
}  
];
```

# Определение массива

```
struct имя_структуры имя_массива[количество]  
[= { {  
    {  
    . . .  
    }  
};
```

# Определение массива

```
struct имя_структуры имя_массива[количество]  
[ = { { значение_01, значение_02, . . . },  
      { значение_11, значение_12, . . . },  
      . . .  
    }];
```

# Определение массива

```
struct имя_структуры имя_массива[количество]  
[ = { { значение_01, значение_02, . . . },  
      { значение_11, значение_12, . . . },  
      . . .  
    }];
```

```
struct Point pp[3] =  
{{1, 1}, {2,2}, {1,2}};
```



# Определение массива

```
struct имя_структуры имя_массива[количество]  
[ = { { значение_01, значение_02, ... },  
      { значение_11, значение_12, ... },  
      ...  
    }];
```

```
struct Point pp[3] =  
{{1, 1}, {2,2}, {1,2}};
```

pp

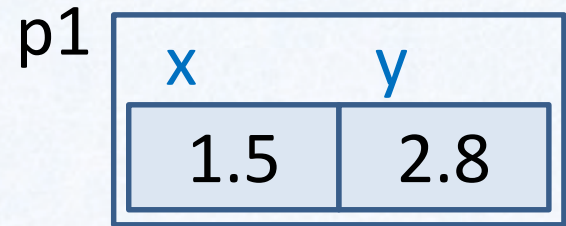
x	y
1.0	1.0
2.0	2.0
1.0	2.0

# Копирование структуры

```
struct Point p1 = {1.5, 2.8};
```

# Копирование структуры

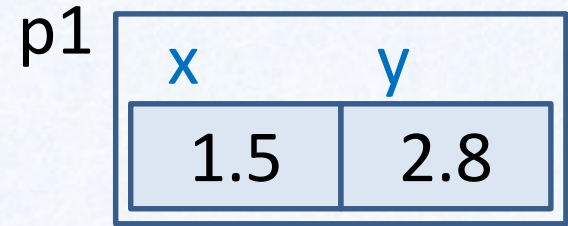
```
struct Point p1 = {1.5, 2.8};
```



# Копирование структуры

```
struct Point p1 = {1.5, 2.8};
```

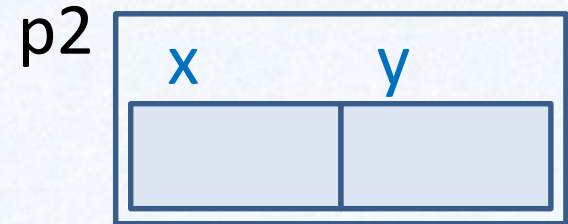
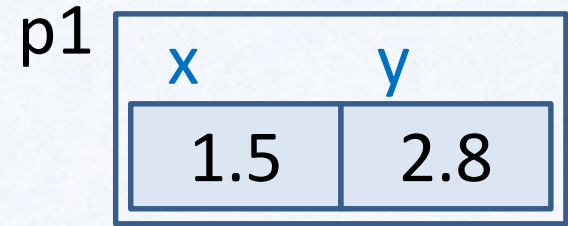
```
struct Point p2 = p1;
```



# Копирование структуры

```
struct Point p1 = {1.5, 2.8};
```

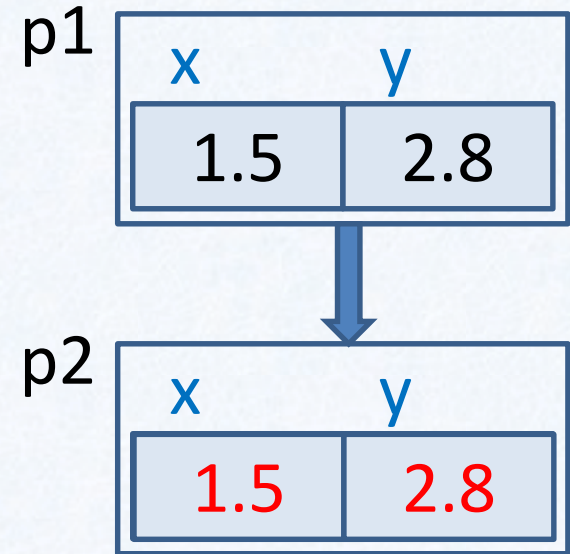
```
struct Point p2 = p1;
```



# Копирование структуры

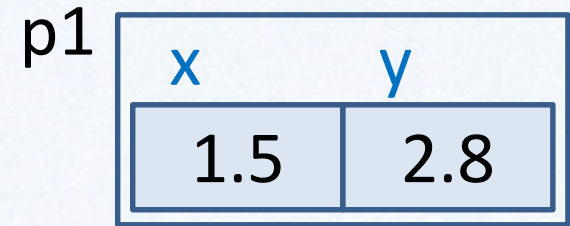
```
struct Point p1 = {1.5, 2.8};
```

```
struct Point p2 = p1;
```



# Присваивание структуры

```
struct Point p1 = {1.5, 2.8};
```

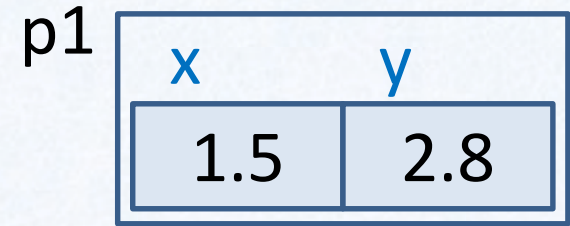




# Присваивание структуры

```
struct Point p1 = {1.5, 2.8};
```

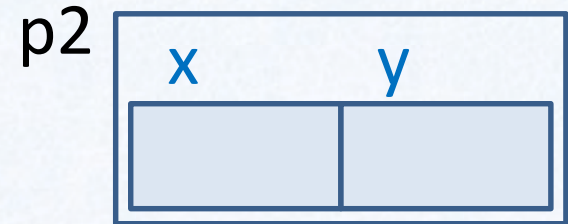
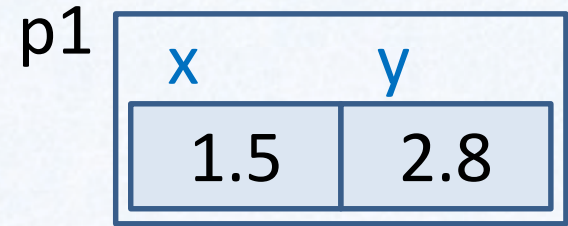
```
struct Point p3;
```



# Присваивание структуры

```
struct Point p1 = {1.5, 2.8};
```

```
struct Point p3;
```

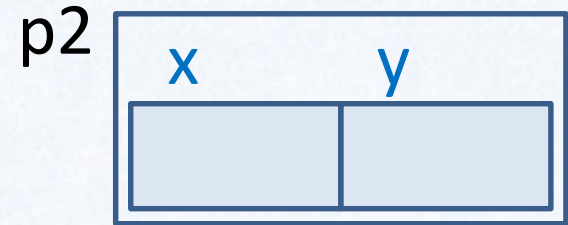
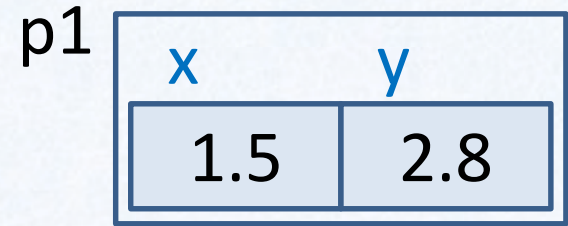


# Присваивание структуры

```
struct Point p1 = {1.5, 2.8};
```

```
struct Point p3;
```

```
p3 = p1;
```

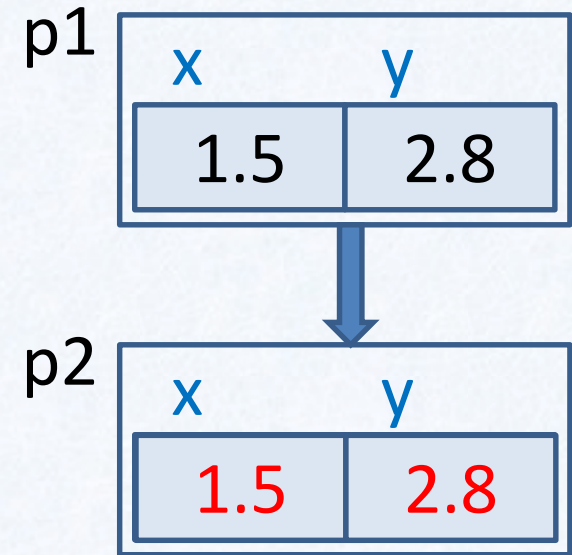


# Присваивание структуры

```
struct Point p1 = {1.5, 2.8};
```

```
struct Point p3;
```

```
p3 = p1;
```

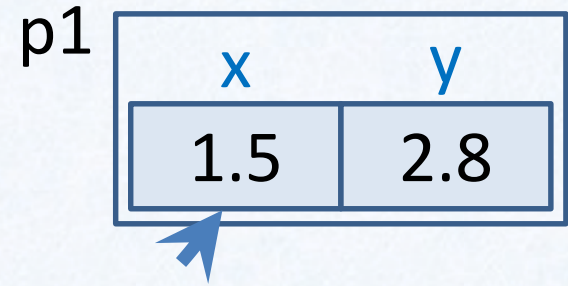


## C4.8 Разыменование структуры

```
struct Point p1 = {1.5, 2.8};
```

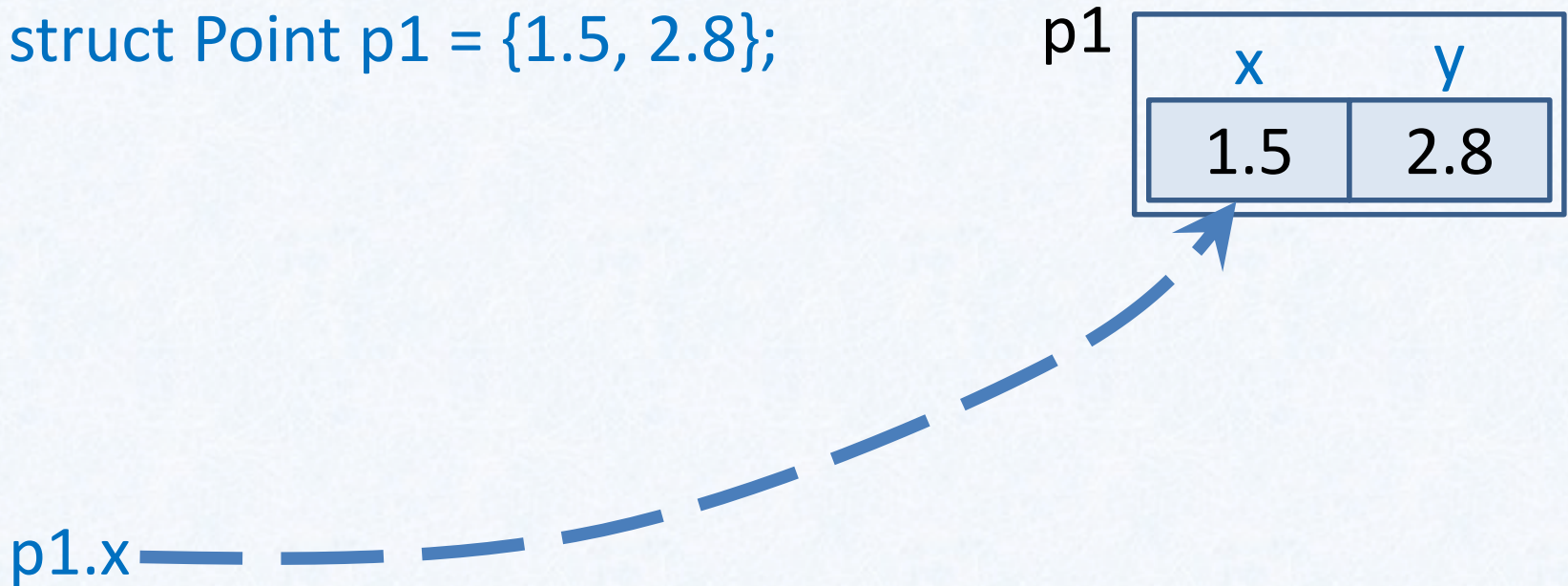
# Разыменование структуры

```
struct Point p1 = {1.5, 2.8};
```



# Разыменование структуры

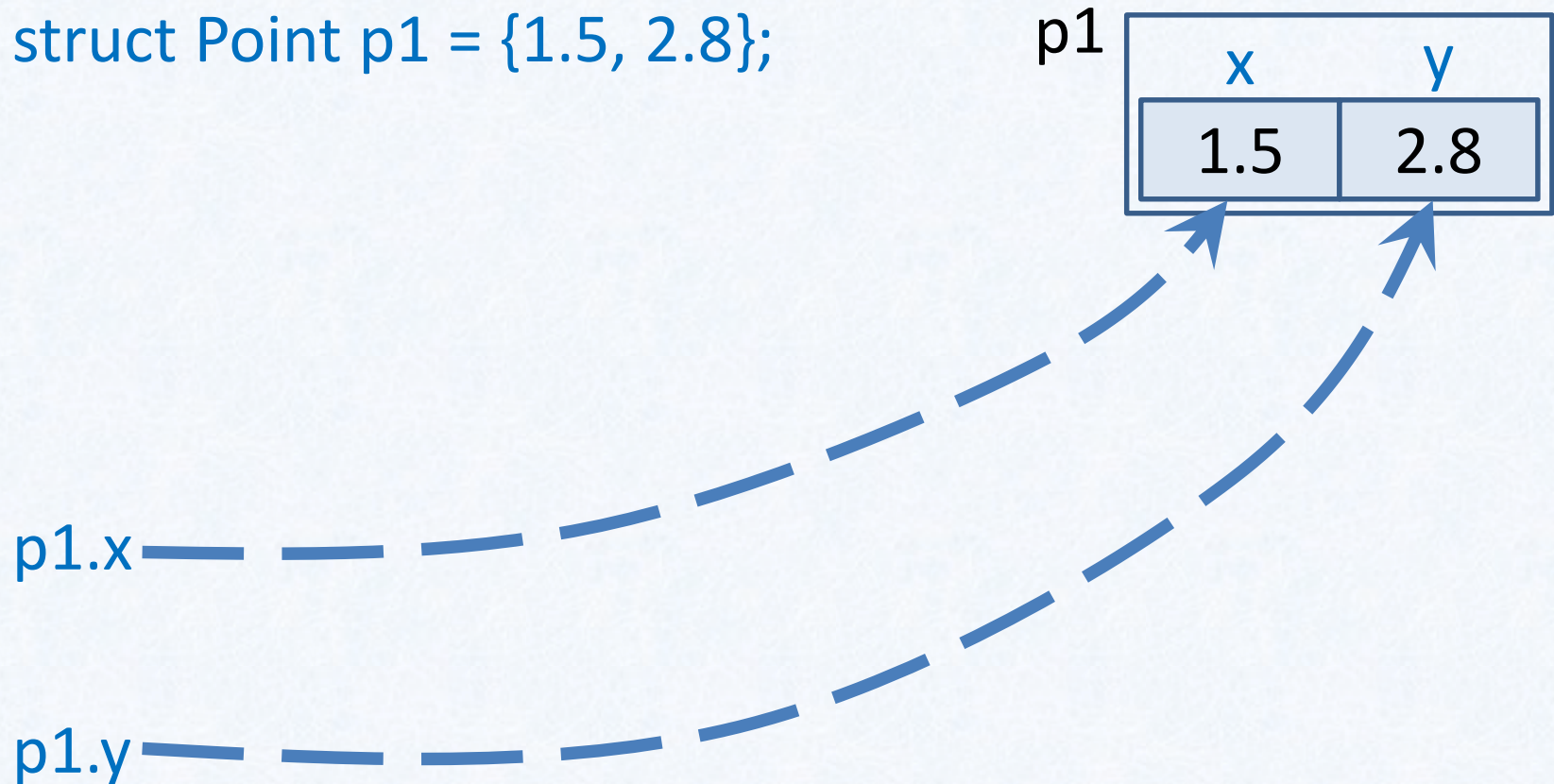
```
struct Point p1 = {1.5, 2.8};
```





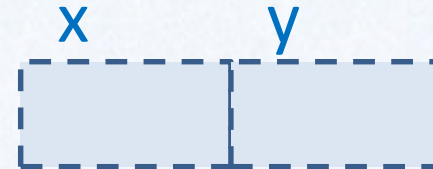
## C4.8 Разыменование структуры

```
struct Point p1 = {1.5, 2.8};
```



# Вложенные структуры

```
struct Point {  
    double x,y;  
};
```



# Вложенные структуры

```
struct Point {  
    double x,y;  
};
```



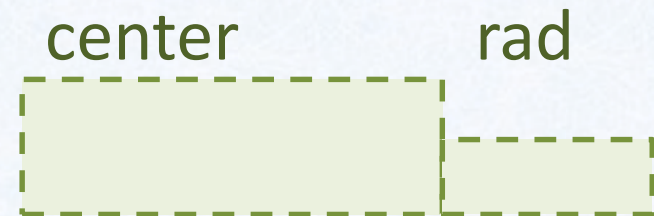
```
struct Circle {  
    struct Point center;  
    double rad;  
};
```

# Вложенные структуры

```
struct Point {  
    double x,y;  
};
```



```
struct Circle {  
    struct Point center;  
    double rad;  
};
```

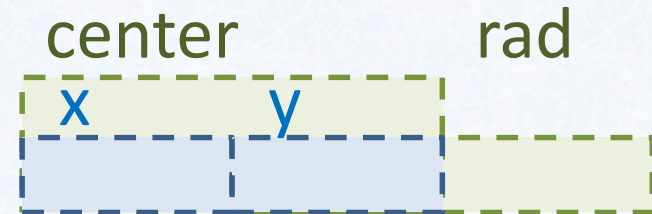


# Вложенные структуры

```
struct Point {  
    double x,y;  
};
```



```
struct Circle {  
    struct Point center;  
    double rad;  
};
```

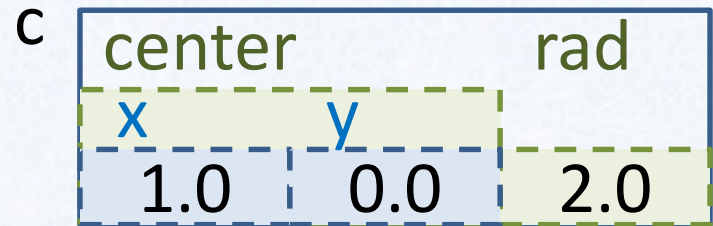


# Вложенные структуры

```
struct Circle c = {{1, 0}, 2};
```

# Вложенные структуры

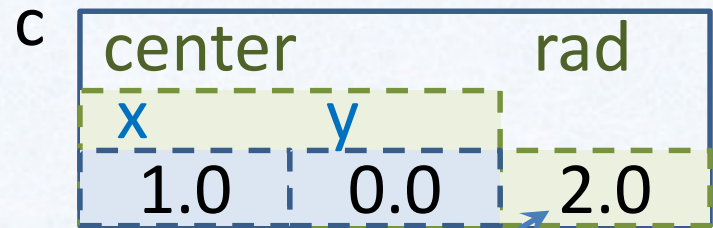
```
struct Circle c = {{1, 0}, 2};
```





# Вложенные структуры

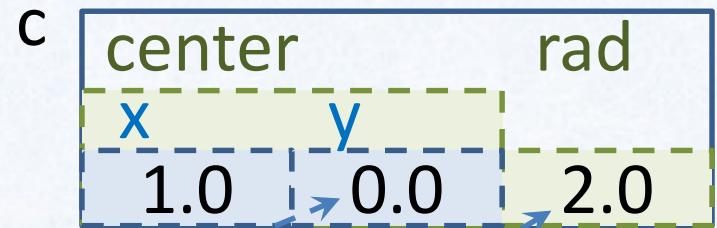
```
struct Circle c = {{1, 0}, 2};
```



c . rad

# Вложенные структуры

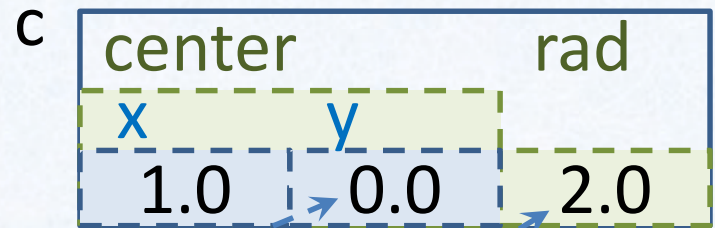
```
struct Circle c = {{1, 0}, 2};
```



c . rad

# Вложенные структуры

```
struct Circle c = {{1, 0}, 2};
```

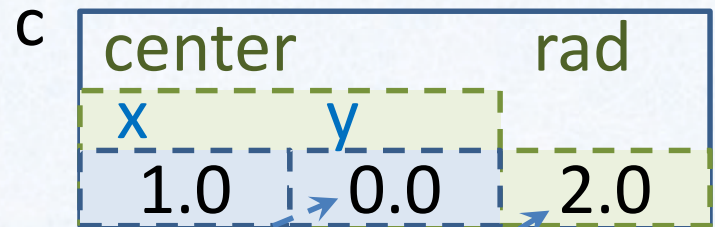


c . center

c . rad

# Вложенные структуры

```
struct Circle c = {{1, 0}, 2};
```



c . center . y

c . rad

# Указатели на структуру

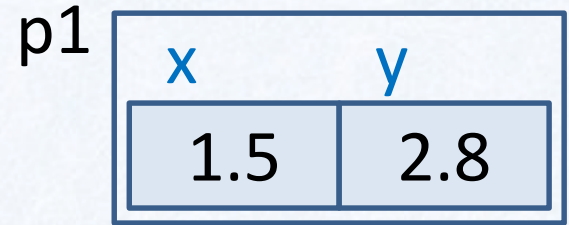
```
struct Point {  
    double x, y;  
};
```

# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};
```

# Указатели на структуру

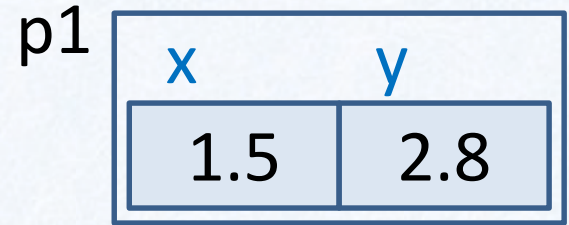
```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};
```





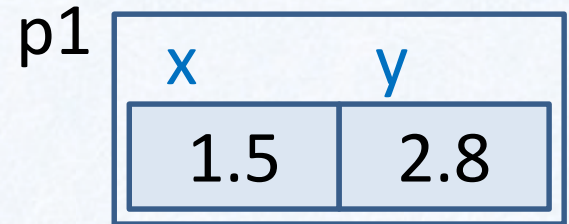
# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr;
```



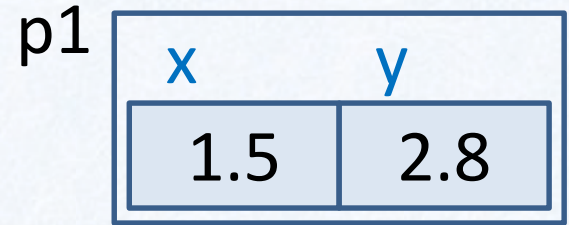
# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr;
```



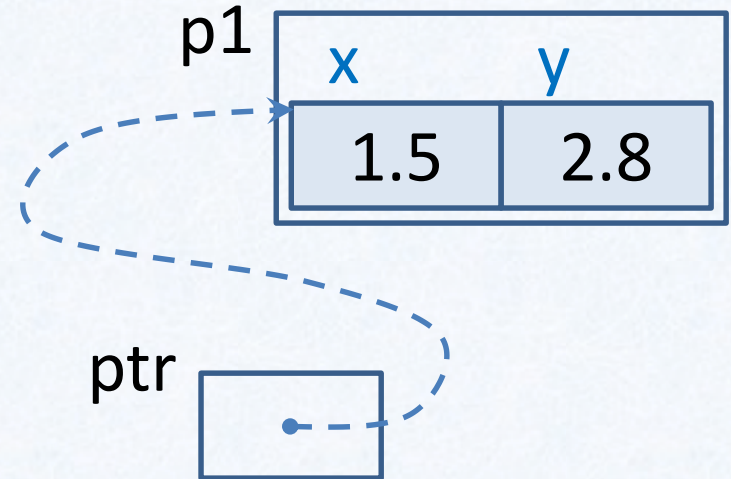
# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr = &p1;
```



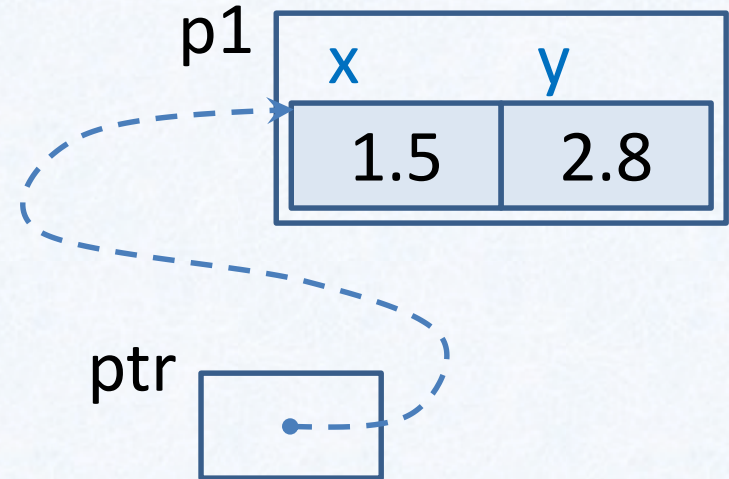
# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr = &p1;
```



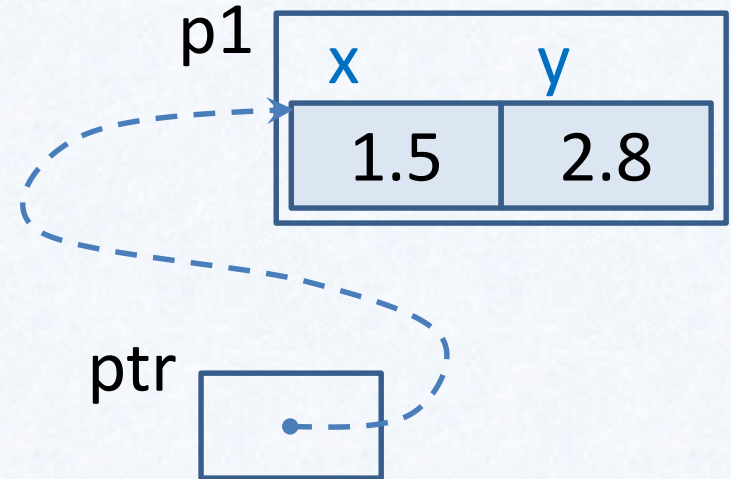
# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr = &p1;  
Разыменование структуры:
```



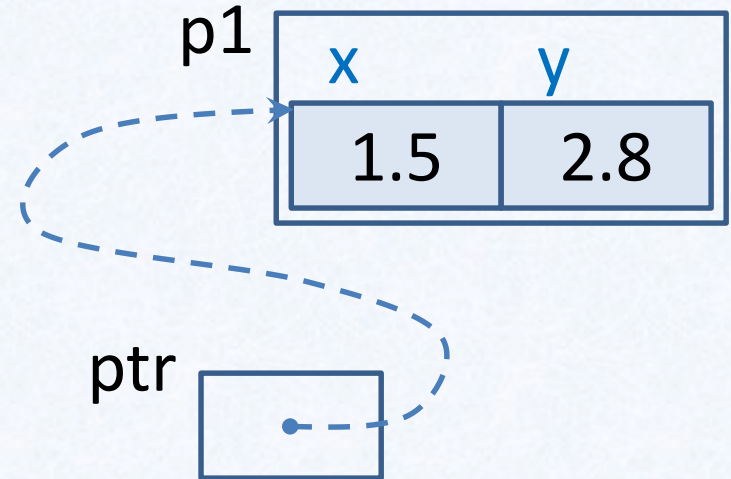
# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr = &p1;  
Разыменование структуры:  
(*ptr)
```



# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr = &p1;  
Разыменование структуры:  
(*ptr).x
```





# Указатели на структуру

```
struct Point {  
    double x, y;  
};  
struct Point p1 = {1.5, 2.8};  
struct Point *ptr = &p1;
```

Разыменование структуры:

`(*ptr).x`                      `ptr -> x`

