

```

// W2_0.cpp : Defines the entry point for the console application.
//           Przekazanie obiektu funkcji przez wartosc i przez wskaznik
//           W pierwszym przypadku funkcje dokonuje zmiany kopii
//           W drygim - samego obiektu

#include "stdafx.h"
#include <iostream>
using namespace std;

class samp;    // referencje zapowiadajaca
void sqr_it(samp ob);
void sqr_it(samp *ob);

class samp
{
    int i;
public:
    samp(int n) {i = n; }
    void set_i(int n) { i = n; }
    int get_i() { return i; }
};

void sqr_it(samp ob)
/*=====
dokonujemy zmiany: skladowa i podnosimy do kwadratu
Przeciez te zmiany - w kopii
=====*/
{
    ob.set_i(ob.get_i()*ob.get_i());
    cout << "kopie obiektu: i = " << ob.get_i() << endl;
}

void sqr_it(samp *ob)
/*=====
dokonujemy zmiany: skladowa i podnosimy do kwadratu
Wskaznik ob wskazuje na obiekt - zmiane samego obiektu
=====*/
{
    ob->set_i(ob->get_i()*ob->get_i());
    cout << "sqr_it: obiekt: i = " << ob->get_i() << endl;
}

```

```
}  
  
int _tmain(int argc, _TCHAR* argv[])  
{  
    samp a(10);  
  
    cout << "obiekt i = " << a.get_i() << endl;  
    //zmieniamy skladowa i  
    sqr_it(a);  
    //zmiany okazali sie tylko w kopii  
    //oryginal sie nie zmienil  
    cout << "obiekt i = " << a.get_i() << endl;  
  
    //przekazanie obiektu po wskaniku: wszystkie zmiany, ktore  
    //sa dokonywane w funkcji nad obiektem, beda widoczne po  
    //wyjsciu z funkcji  
    sqr_it(&a);  
    cout << "obiekt i = " << a.get_i() << endl;  
  
    system("pause");  
  
    return 0;  
}
```