



## Jeszcze jeden kwadrat?

```
#typ_logiczny #bool
#predykat     #true/false
```

No i co? Wracamy do obliczania kwadratów liczb naturalnych? Nie, tym razem napiszemy program, który sprawdza, czy dana dodatnia liczba całkowita jest kwadratem liczby naturalnej i wypisuje stosowny komunikat. Na przykład dla liczby 16 program powinien wypisać **TAK**, a dla 30 powinien wypisać **NIE**.

Ponieważ wspomnieliśmy co nieco o funkcjach, zatem napiszemy własną funkcję, która będzie sprawdzać, czy jej argument spełnia powyższy warunek. Funkcja – jak łatwo się domyślić – powinna zwrócić jedną z wartości prawda/fałsz (**true/false**, wymawiaj: *tru/fols*).<sup>\*</sup> Taki typ danych nosi nazwę *typu logicznego* i na cześć zasłużonego dziewiętnastowiecznego matematyka angielskiego George’a Boole’a nazywa się `bool` (wymawiaj: *bul*).

Jak będziemy sprawdzać, czy liczba, dajmy na to  $m$ , jest kwadratem? Zrobimy to metodą *na palę*: weźmiemy pomocniczą zmienną  $p$ , na początku równą 1, i będziemy ją tak długo zwiększać o 1, aż uzyskamy sytuację, gdy  $p^2 = m$  (wtedy funkcja zwróci wartość `true`) albo  $p^2 > m$  (i wtedy funkcja zwróci wartość `false`, bo już nie mamy szansy na sukces).

Chyba możemy po prostu napisać cały program z funkcją `is_square()` (z ang. *jest kwadratem*, wymawiaj: *iz skter*):<sup>†</sup>

```
#include <bits/stdc++.h>
using namespace std;

bool is_square(int m)
{
    int p{ 1 };
    while(p * p <= m)
    {
        if(p * p == m)
            return true;
        p++;
    }
    return false;
}
```

---

<sup>\*</sup>W logice matematycznej taka funkcja nosi nazwę *predykatu*.

<sup>†</sup>Zauważ użycie znaku podkreślenia `_` w nazwie funkcji w celu oddzielenia dwóch wyrazów.

```
int main()
{
    int n;
    cin >> n;
    if(is_square(n))
        cout << "TAK\n";
    else
        cout << "NIE\n";
    return 0;
}
```

Jak widać, pętla `while` w funkcji „kręci się”, póki  $p^2 \leq m$ . Jeśli trafiliśmy z wartością  $p$ , wtedy od razu zwracana jest wartość `true`. Instrukcja `return` jest przykładem instrukcji skoku i powoduje natychmiastowe wyjście z funkcji i powrót do miejsca, gdzie funkcja została wywołana.

Jeśli pętla obejdzie spokojnie do końca, wtedy zwracana jest wartość `false`, gdyż osiągnięto  $p^2 > n$ .