

0.1 Parametryzacja szablonów

Parametry szablonu są specyfikowane na dwa sposoby:

1. *parametry szablonu* – wyrażnie wspomniane jako parametry w deklaracji szablonu
2. *nazwy zależne* - wywnioskowane z użycia parametrów w definicji szablonu

W *C++* nazwa nie może być użyta bez wcześniejszej deklaracji. To wymaga od użytkownika ostrożnego traktowania definicji szablonów. Np. w definicji funkcji `kwadrat` nie ma widocznej deklaracji symbolu `*`. Jednak, podczas inicjalizacji szablonu `kwadrat<int>` kompilator może sprowadzić symbol `*` do (wbudowanego) operatora mnożenia dla wartości `int`. Dla wywołania `kwadrat(zespolona(2.0))`, operator `*` zostałby rozwiązany do (zdefiniowanego przez użytkownika) operatora mnożenia dla wartości `zespolona`. Symbol `*` jest więc *nazwą zależną* w definicji funkcji `kwadrat`. Oznacza to, że jest to ukryty parametr definicji szablonu. Możemy uczynić z operacji mnożenia formalny parametr:

```
template<typename Multiply , typename T>
T square(T x) {
    return Multiply() (x,x);
}
```

Pod-wyrażenie `Multiply()` tworzy obiekt funkcji, który wprowadza operacje mnożenia wartości typu `T`. Pojęcie *nazw zależnych* pomaga utrzymać liczbę jawnych argumentów.