# Interfejsy

1. Dla figur z ćwiczeń 5.1 i 5.2 zaimplementować IEquatable<T>, gdzie dla każdego typu pochodnego figury T to ten sam typ (np: Koło : Figura, IEquatable<Koło>)
   1. możemy porównywać środki lub nie
   2. możemy wymagać zgodności dokładnie tych samych boków albo pozwolić na zgodność boków cyklicznie permutowanych (obróconej figury)
2. W klasie Prostokąt wprowadzić wielkość double Orientacja - w seterze upewniamy się, że jest nieujemne i bierzemy modulo Pi; Zdefiniować porównanie (ćw 6.1) tak, aby brało pod uwagę tę orientację
   1. znowu możemy pozwolić na zamianę boków, czyli gdy this.Orientacja == that.Orientacja + Pi/2 modulo PI, oraz (this.A,this.B) == (that.B, that.A) to prostokąty też są równe
3. W klasie Trójkąt wprowadzić wielkość double Orientacja - w seterze upewniamy się, że jest nieujemne i bierzemy modulo Pi; Zdefiniować porównanie (ćw 6.1) tak, aby brało pod uwagę tę orientację
   1. możemy sprawdzać przypadek identycznej długości ale różnie nazwanych boków, ale jest to dość skomplikowane
4. Dla klasy Punkt z zajęć lub klasy Wektor z ćwiczenia 2.2A zaimplementować IComparable<T> (gdzie T to odpowiednio Punkt albo Wektor)
   1. porównujemy po kolejnych wymiarach i jeśli któryś z porównywanych obiektów ma mniejszy wymiar to traktujemy brakujące wielkości jako -Nieskończoność (więc wektor mniejszego wymiaru jest zawsze mniejszy, jeżeli początkowe współrzędne są równe)
   2. porównujemy odległość od 0