# **POLİFORMİZM**

Polimorfizm kelimesinin kökenine indiğimizde Yunancada ‘’poly’’(çoklu,çok) ve ‘’morphe’’(form) kelimelerinin birleşiminden meydana geldiğini görürüz. Türkçeye bir nesnenin birçok formda olabileceği şeklinde çevrilse de programlama dillerinden biri olan C++ dili için bir işi yapabilen farklı metodlar olarak bu kavramı çevirebiliriz.

Amacımız bir fonksiyonu farklı şekillerde çalıştırabilmektir. Nesne yönelimli özelliği olmayan dillerde, bir fonksiyon tek bir isimle bir kere tanımlanabilir. Örneğin aşağıdaki dikdörtgenin alanını hesaplayan fonksiyonu ele alalım:

int alan(int a, int b){

   return a\*b;

}

Yukarıdaki fonksiyon gayet başarılı bir şekilde a ve b kenarları verilen dikdörtgenin alanını hesaplar. Ancak aynı alan hesabını örneğin float sayılar için düzenlemek istesek ve verilen değere göre çalışmasını istesek:

float alan (float a, float b){

   return a\*b;

}

Şeklinde yazılan fonksiyon, aldığı parametreler itibariyle, farklı bir fonksiyon iken, nesne yönelimli özelliği bulunmayan dillerde, bu iki fonksiyonu aynı anda kodlamamız mümkün olmaz.

Benzer şekilde, her kare bir dikdörtgendir mantığıyla, tek kenar uzunluğu verildiğinde alan hesaplamak istersek

int alan(int a){

   return a\*a;

}

Fonksiyonu, bir karenin alanını başarılı bir şekilde hesaplayabilirken, daha önceden tanımlı olan fonksiyonlarla çakışmaktadır.

Bir işi farklı kodlarla farklı işlemlerle yapabilme yetisine farklı örnekler de verebiliriz fakat temal anlamda polimorfizm bu temelde aynı işi farklı kodlarla yapabilmeye denir.