NYP 7

1. İkiye Bölme (Bisection) Yöntemi

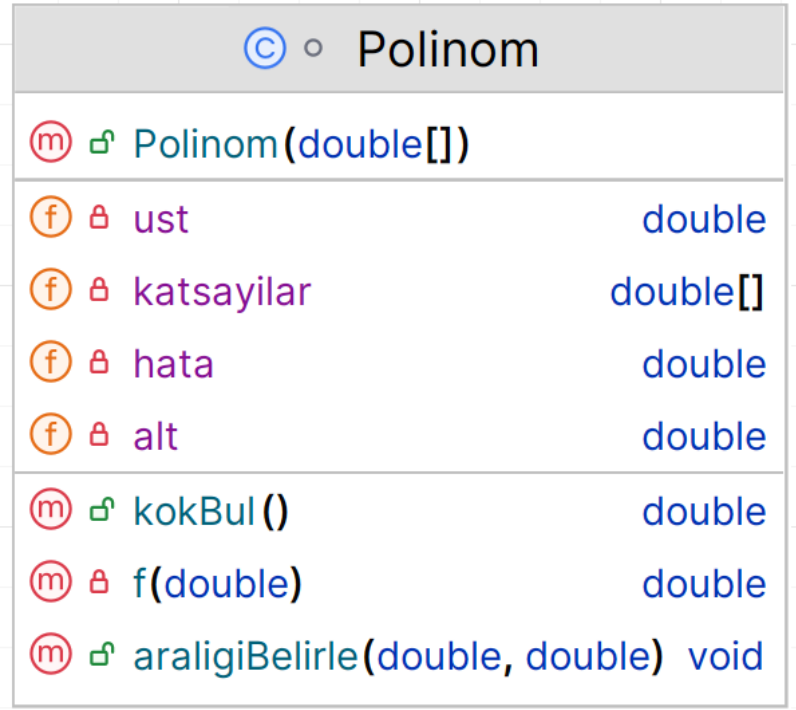
İkiye bölme yöntemi, kökün belirli bir aralıkta bulunduğunu bildiğimiz durumlarda kullanılır. Her adımda aralık ikiye bölünerek kök bulunmaya çalışılır.

Formül:

$$c = \frac{a + b}{2}$$

Burada:

- a ve b aralığın uç noktalarıdır.
- c yeni orta nokta olarak hesaplanır.
- Eğer $f(c) = 0$, kök tam olarak bulunmuş olur.
- Eğer $f(a) \cdot f(c) < 0$ ise kök $[a, c]$ aralığında bulunur, aksi halde $[c, b]$ aralığında aranır.




```
public class Main {
    public static void main(String[] args) {
        double[] katsayilar = {1,0,-4}; // f(x) = x²-4
        Polinom p = new Polinom(katsayilar);
        p.araligiBelirle(0, 8);
        double kok=p.kokBul();
        System.out.println(kok);
    }
}
```

```python
class Ogrenci:
    okul = "Trakya Üniversitesi"
    def __init__(self, isim, numara):
        self.isim = isim
        self.numara = numara
    def bilgileri_goster(self):
        return f"İsim: {self.isim}, Numara: {self.numara}, Okul: {Ogrenci.okul}"

# Nesneler oluşturma
ogrenci1 = Ogrenci("Ali", 101)
ogrenci2 = Ogrenci("Ayşe", 102)

print(ogrenci1.bilgileri_goster())
print(ogrenci2.bilgileri_goster())
```

Python’da Statik Sınıf Yapısı

```
class Ornek:
    @staticmethod
    def toplama(a, b):
        return a + b
```

```python
class Kitap:
    def __init__(self, ad, yazar):
        self.ad = ad
        self.yazar = yazar

    def __str__(self):
        return f"Kitap: {self.ad} - Yazar: {self.yazar}"

kitap = Kitap("Suç ve Ceza", "Dostoyevski")
print(kitap)
```

```
def main():
    k1 = KarmasikSayi(1.0, 2.0)
    k2 = KarmasikSayi(3.0, 4.0)
    ks = KarmasikHesaplama()
    ks.topla(k1, k2)
    ks.toplamiYaz()

if __name__ == "__main__":
    main()
```

```python
class KarmasikSayi:
    def __init__(self, real, imag):
        self.real = real
        self.imag = imag

    def __add__(self, other):
        if isinstance(other, KarmasikSayi):
            return KarmasikSayi(self.real + other.real, self.imag + other.imag)
        return NotImplemented

    def __str__(self):
        return f"{self.real} + {self.imag}j"


class KarmasikHesaplama:
    def __init__(self):
        self.toplam = None
    def topla(self, k1, k2):
        self.toplam = k1 + k2

    def toplamiYaz(self):
        print("Karmaşık Toplam:", self.toplam)
```

Özel metodlar

init, str, add,sub,mul,truediv,eq,lt,gt,eq...