

Cyclomatic Complexity

Yazılan kodun karmaşıklığını ölçmek için kullanılan bir terimdir. Bir programın kaynak kodu aracılığıyla doğrusal olarak bağımsız yolların sayısının nicel bir ölçüsüdür. Yazılan kod ne kadar karmaşık olursa o kadar sürdürülebilirliği zor olur.

İyi yazılmış bir programın karmaşıklığının düşük olması beklenir.

Doğru bir hesaplama çıkarabilmek için programın şu sorulara cevap vermesi beklenir :

1. Program test edilebilir mi?
2. Program herkes tarafından anlaşılabilir mi?
3. Program yeterince güvenilir mi?

Bu soruların cevaplarını bulmak için çeşitli yöntemler kullanılmaktadır.

Yöntemler nedir? Nasıl kullanılır?

Bir programdaki ifadeler düğüm olarak temsil edilir ve bir ifadeden diğerine kontrol yolları Kenarlar tarafından temsil edilir.

Hesaplama Formülü:

$$\text{Cyclomatic Complexity} = E - N + 2 \quad (E = \text{Kenar sayısı}, N = \text{Düğüm sayısı.})$$

Cyclomatic Complexity Hesaplama Örneği :

ÖRNEK2

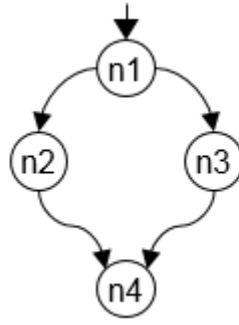


Düğüm Sayısı (Kırmızı noktalar) = 14

Kenar Sayısı (Mavi Çizgiler) = 15

Yani Cyclomatic Karmaşıklık $\Rightarrow N - E + 2 = (14 - 15) + 2 = 3$

```
if expression1 then
    statement2
else
    statement3
end if
statement4
```



$$V(G) = 4 - 4 + 2 = 2$$

Yukarıdaki kod bloğunda expression1 deyimi grafta bulunan n1 düğümüne denk düşer. n1 düğümü if bloğunda görüldüğü üzere n2 ve n3 düğümlerine ulaşabilir. Program kodu son olarak yoluna devam eder. Burada toplamda 4 adet ayrıt yani düğümleri birbirlerine bağlayan ok-bağlantı çubuğu bulunur. Aynı zamanda her bir ifadeyi ihtiva eden 4 adette düğüm bulunur. Cyclomatic complexity bağıntısına göre karmaşıklık değeri 2 dir. Bu sonuç enstitüye göre kodun riskli olmayan bir modül olduğunu gösterir.

HAZIRLAYAN : ÖMER CENGİZ 16260056