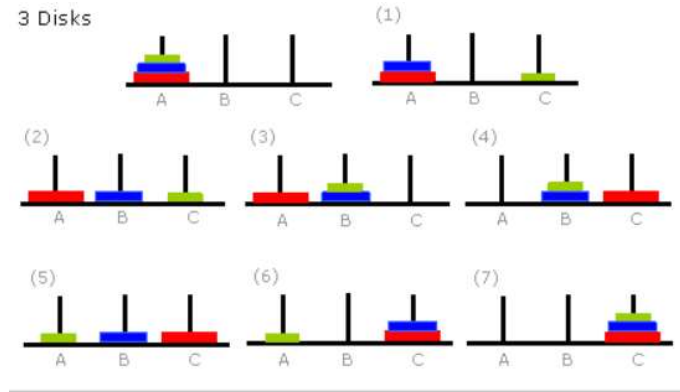


 SELÇUK ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ	T.C. SELÇUK ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ STAJ RAPORU	Tarih Sayfa No:	12/07/2021 30
---	--	------------------------	----------------------

Üç diskin ortada toplanması 7 hamle almıştır, bundan sonraki 1 hamle, dördüncü diske sağa almaktır. Sonraki hamle ise, 7 hamlede üç diski, bu en büyük diskin üzerine almaktır. Yani yapılan işlem sayısı $7+1+7=15$ 'tir.



Resim 13.2 3 disk için Hanoi Kuleleri çözümü

Örneğin 4 disk problemini (Resim?.1) 3 disk ile göstermek istersek işlem sayısını $(3+(1)+3)+1+(3+(1)+3)$ olarak gösterebiliriz (Resim?.2).

Yine bu durum 3 diskli bir çözüm için $(1+(1)+1)+1+(1+(1)+1)$ olarak gösterilebilir.

Bu durumu formülize edersek;

3 platform için n sayıda diskin çözüm formülü $2^n - 1$ olarak gösterilir.

Kodumu bu algoritma yapısına uygun yazarak fazla hamlelerden kaçınmış olacağım.

STAJ SORUMLUSU
Onay (İmza & Kaşe)
..... / / 20.....

 SELÇUK ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ	T.C. SELÇUK ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ STAJ RAPORU		Tarih Sayfa No:	13/07/2021 31
---	---	--	------------------------	----------------------

Hanoi kuleleri konsol uygulaması için yaptığım araştırmalardan ve kurduğum algoritma mantığından sonra kod yazımını tamamladım.

```

public class HanoiKuleleri {
    int sayac=1;
    public void bilgiYazdir(int n, String platf_1, String platf_2, String platf_3){
        System.out.println(sayac+". hamle: "+n+". disk "+ platf_1+" platformdan "+platf_3+" platforma taşındı.");
        sayac++;
    }
    public void diskYerleşim(int n, String platf_1,
        String platf_2, String platf_3)
    {
        if (n == 1)
        {
            bilgiYazdir(n, platf_1, platf_2, platf_3);
            return;
        }
        else{

            diskYerleşim(n - 1, platf_1, platf_3, platf_2);
            bilgiYazdir(n, platf_1, platf_2, platf_3);
        }

        diskYerleşim(n - 1, platf_2, platf_1, platf_3);
    }
    public static void main(String args[])
    {
        HanoiKuleleri hanoi = new HanoiKuleleri();
        hanoi.diskYerleşim(4, "ilk", "orta", "son");
    }
}

```

Resim 14.1 HanoiKuleleri Sınıfı

Öncelikle HanoiKuleleri adında bir sınıf tanımladım. Bu sınıf içerisinde asıl işi yapan metodum public void tipinde tanımladığım değer döndürmeyen diskYerleşim() (Resim 14.1) metodudur. Bu metod int tipinde bir parametre ve string tipinde 3 parametre alır.

Bu parametrelerden integer tipindeki n sayısı mevcut disk sayısını belirlemektedir. Temel amacım özyinelemeli fonksiyon yapısı oluşturmaktır. Bu n sayısı ile hareket sırası olan disk devamlı tutuyor olacak. Kontrol yapısında da geçerli yapılabilir hareketlerin hepsini özyineleme sayesinde gerçekleştirmekteyim.

Yine her adımımı bir adet int, 3 adet string değişken alan public void tipinde tanımlanan değer döndürmeyen bilgiYazdir() fonksiyonu ile yazdırdım. sayac değişkeni ile her adımımı numaralandırdım.

Test kısmında da sınıftan hanoi isimli bir nesne türettim. Bu nesneye çözüm için gerekli olan disk sayısını ve platform isimleri girilebilir. Ben platformları “ilk”, “orta” ve “son” platform olarak adlandırdım.

STAJ SORUMLUSU
Onay (İmza & Kaşe)
..... / / 20.....

 SELÇUK ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ	T.C. SELÇUK ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ STAJ RAPORU	Tarih Sayfa No:	13/07/2021 32
---	--	------------------------	----------------------

```

--- exec-maven-plugin:3.0.0:exec (default-cli) @ HanoiKuleleri
1. hamle: 1. disk ilk platformdan orta platforma taşındı.
2. hamle: 2. disk ilk platformdan son platforma taşındı.
3. hamle: 1. disk orta platformdan son platforma taşındı.
4. hamle: 3. disk ilk platformdan orta platforma taşındı.
5. hamle: 1. disk son platformdan ilk platforma taşındı.
6. hamle: 2. disk son platformdan orta platforma taşındı.
7. hamle: 1. disk ilk platformdan orta platforma taşındı.
8. hamle: 4. disk ilk platformdan son platforma taşındı.
9. hamle: 1. disk orta platformdan son platforma taşındı.
10. hamle: 2. disk orta platformdan ilk platforma taşındı.
11. hamle: 1. disk son platformdan ilk platforma taşındı.
12. hamle: 3. disk orta platformdan son platforma taşındı.
13. hamle: 1. disk ilk platformdan orta platforma taşındı.
14. hamle: 2. disk ilk platformdan son platforma taşındı.
15. hamle: 1. disk orta platformdan son platforma taşındı.
-----

```

Resim 14.2 Hanoi Kuleleri Ekran Çıktısı

Bu adımları sırayla gerçekleştirdiğimizde problem çözülmüş olmaktadır. Resim 13.1'deki 4 diskli Hanoi Kulesi çözümü ile sonucum eşleşmektedir.

STAJ SORUMLUSU
Onay (İmza & Kaşe)
..... / / 20.....