      \* STATE: Substring gibi konular için optimal daha hızlı çözüm sağlar.

         \* state ile bir konumdan başka bir konuma yani state ya geçeriz yada kendi statemize geri döneriz

         \* otomata yapısı statelerden oluşur. Birden fazla state olabilir.

         \* Otomata ile search,compiler gibi uygulama yapılabilir.

         \* Otomata sadeece ve sadece sayma işlemi yapamaz: Çünkü otomata sayma yetisi sınırlıdır. Otomata içinde dögü tam anlamıylada yok.

         \* Sadece ve sadece "Push otomata " ile bu sorun belirli bir seviyeye kadar optimum şekilde çözülür.

         \*/

        // ınput karakterei = alfabe

      int state =0;

      int[,] q = new int[4,2]; // 4--> state sayısı, 2--> input sayısı

      string otomat = "111111010111";

      q[0,0] = 0;

      q[0,1] = 1;

      q[1,0] = 0;

      q[1,1] = 2;

      q[2,0] = 3;

      q[2,1] = 2;

      q[3,0] = 3; // buradaki 3 değeri dead state'i temsil eder.

      q[3,1] = 3;

      for (int i = 0; i < otomat.Length; i++)

      {

        state = q[state,otomat[i]-'0']; // 0 yazmamızın sebebi "010101"

        // string içerisindeki her karakteri alıp ASCII karşılığına göre işlem yaparız.

        //  Bu işlem karakter olarak saklanan rakamları gerçek değerlerine dönüştürür 0--> 48, 1--> 49

        // Harflerde ise a değerini çıkarıp işlem yaparsın. a-->97 = 0, b-->98  =1 gibi işlem yaparız.

        // Eğer 0 yazmazsak  48 ve 492u sizi indeksi olarak kullanmaya çalışır ve sonucundada  IndexOutOfRangeException hatası alırsın.

      }

      if(state == 3) Console.WriteLine("olumsuz");

      else Console.WriteLine("olumlu");

      // def --> string içerisinde olup olmadığını anlayan Automata

      int states =0;

      int[,] a = new int[4,30];

      string str="abcdefghklmnoprstuvyzfefeofe";

      // a,b,c,d,e,f

      // 0,1,2,3,4,5

      for (int i = 0; i < 30; i++)

      {

        q[0,i] = 0; // buradaki i değeri  buraya gelecek olan input değerinin ASCII karşılışıdır.

            //

      }

      q[0,3] = 1;

      q[1,4] = 2;

      q[1,3] = 3;

      q[2,5] = 2;

      q[2,3] = 3;

      for (int i = 0; i < 30; i++)

      {

            q[3, i] = 3;

      }

      for (int i = 0; i < str.Length; i++)

      {

          states = q[states, str[i] - 'a'];

      }

      if (states == 3) Console.WriteLine("olumlu");

      else Console.WriteLine("olumsuz");

        // 1 ve 0 lardan oluşan bir string geliyor. Decimal halinin mod 5 e göre kalanı bul.

        int[,] s = new int[5, 2]; // 5 state var ve 2 input sayısı(0 ve 1) var

        s[0, 0] = 0;

        s[0, 1] = 1;

        s[1, 0] = 2;

        s[1, 1] = 3;

        s[2, 0] = 4;

        s[2,1] = 0;

        s[3, 0] = 1;

        s[3, 1] = 2;

        s[4, 0] = 3;

        s[4, 1] = 4;

        int state1 = 0;

        string input = "101010101010";

        for (int i = 0; i < input.Length; i++)

        {

            state1 = s[state1, input[i] - '0'];

        }

        Console.WriteLine(state1); // elde edilen kalan değeri

    }

}