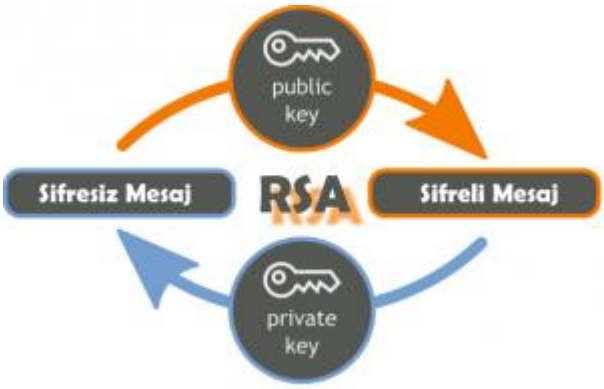
RSA ALGORİTMASI

KRİPTOLOJİ PROJE ÖDEVİ

ZEYNEP TORUN – G151210307
MUHAMMET RAŞİT TOSUN - B171210372

RSA Algoritması

RSA algoritması 1978 yılında Ron Rivest, Adi Shamir, Leonard Adleman kişileri tarafından bulunup, ismini bu kişilerin soyadlarının baş harflerinden almıştır. Güvenliği tam sayıları çarpanlarına ayırmanın algoritmik zorluğuna dayanan bir tür açık anahtarlı şifreleme yöntemidir. Simetrik şifreleme yapan algoritmalara göre güvenlik açısından daha başarılıdır ancak hız ve donanımsal uygunluk bakımından dezavantajı vardır. DH(Diffie-Helman) ve RSA algoritmaları asimetrik şifreleme algoritmaları gurubunda yer alır.



Şifreleme ve elektronik imza uygulamalarında kullanılmaktadır.

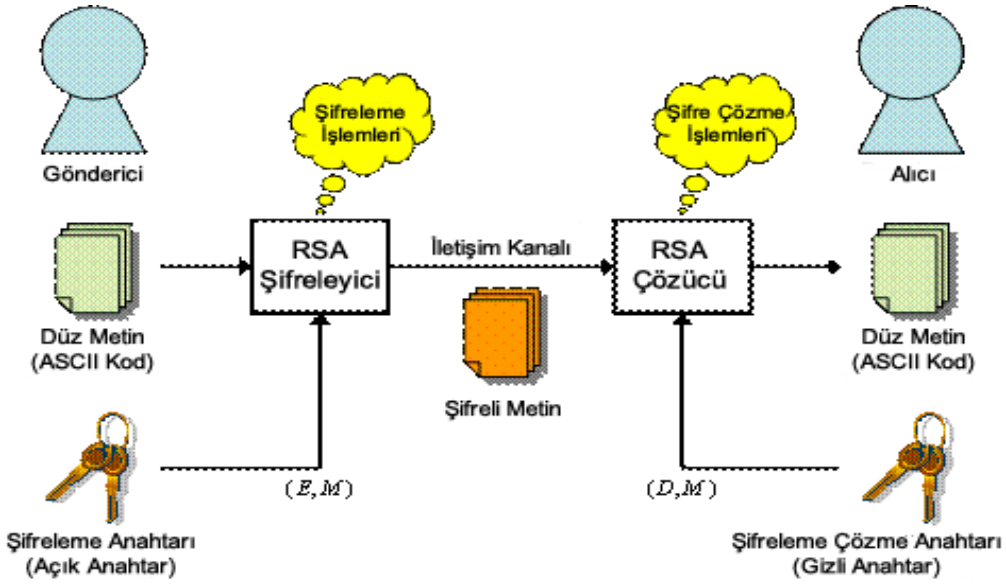
Bu algoritma için iki anahtara ihtiyaç vardır. Şifreli mesaj gönderebilmek için kullanılan açık anahtar ve mesajı alan tarafın şifreli metni açabilmesi için gönderilen şifreli metnin açılabilmesi için ise gizli anahtara sahip olması gerekir.

DEZAVANTAJLARI

- Her kullanıcı kendine ait bir şifre tutmasıyla bütün gizlenmiş metinleri tek anahtarla çözme imkânına sahiptir. Bu da anahtarın kaybolması durumunda veya başka birinin ele geçirmesi durumunda büyük sıkıntılara yol açmaktadır.
- Anahtarların değiş tokuş edilmesi gerektiğinden dolayı anahtarı karşı tarafa iletmek için bir ağ kullanma zorunluluğu vardır. Bu da ağda ekstra güvenlik önlemi almayı gerektirmektedir.

Örneğin n tane kullanıcı olması durumunda n−1 adet şifre oluşturmak gerekli ve bu sistemde tutulmalıdır. Bu da ekstra bellek alanı tutacağından bu da bir dezavantaj olarak görülebilir.

ÇALIŞMA PRENSİBİ



Yeterince büyük iki adet asal sayı seçilir. (Seçili sayıların büyük olmasıyla asal olup olmadığının kontrolünü yapmak zor olacağından bu kısımda Fermat Teoremi kullanılabilir.)

Örneğin; p ve q olsun.

Bu iki asal sayının çarpımı hem açık hem de kapalı şifreler için taban olarak(mod) seçilir.

$$n = p * q$$

Yine bu iki asal sayı için Totient Fonksiyonu(T(n)) hesaplanır.

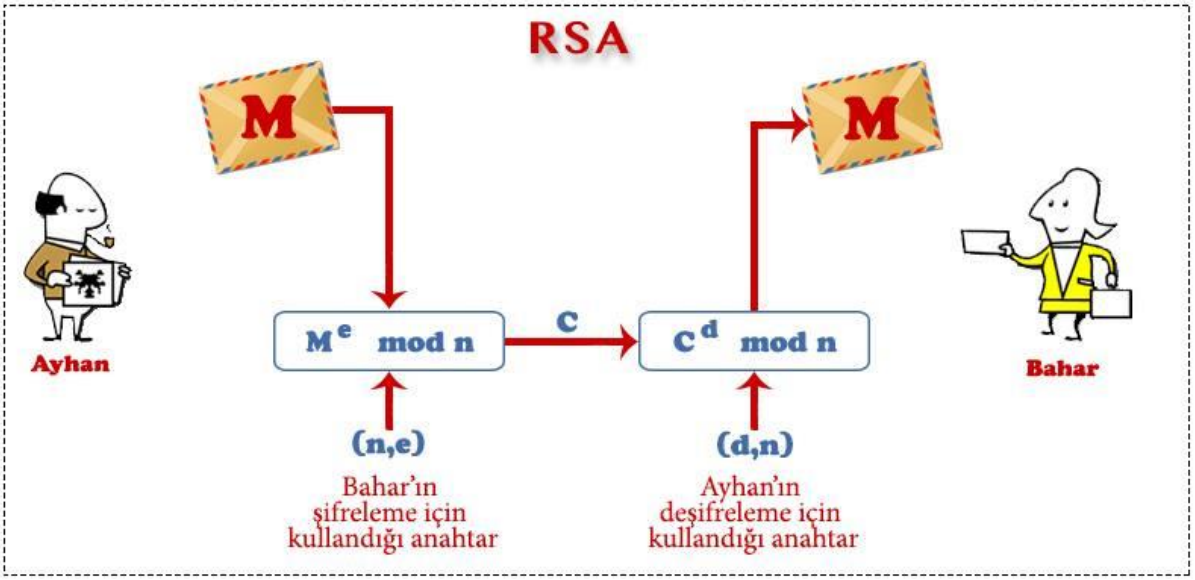
$$T(n) = (p-1)(q-1)$$

Hesaplanan T(n) değeri ile aralarında asal olan bir açık anahtar(e) belirlenir.

$$1 < e < T(n) \text{ olmalıdır.}$$

Aşağıdaki denkliğe sahip kapalı anahtar(d) seçilir. Bu anahtar öyle bir sayı olmalıdır ki e sayısı ile çarpımının T(n) değerine göre mod alındığında 1 çıkmalı yani genişletilmiş öklit algoritmasından yararlanılarak T(n) mod değerine göre e anahtarının tersi d anahtarı olarak belirlenmelidir.

$$de \equiv 1 \pmod{T(n)}$$



Tüm bu işlemlerden sonra şifrelenecek metnin her bir karakterinin ASCII kod değerlerine göre

$$c = m^e \bmod(n)$$

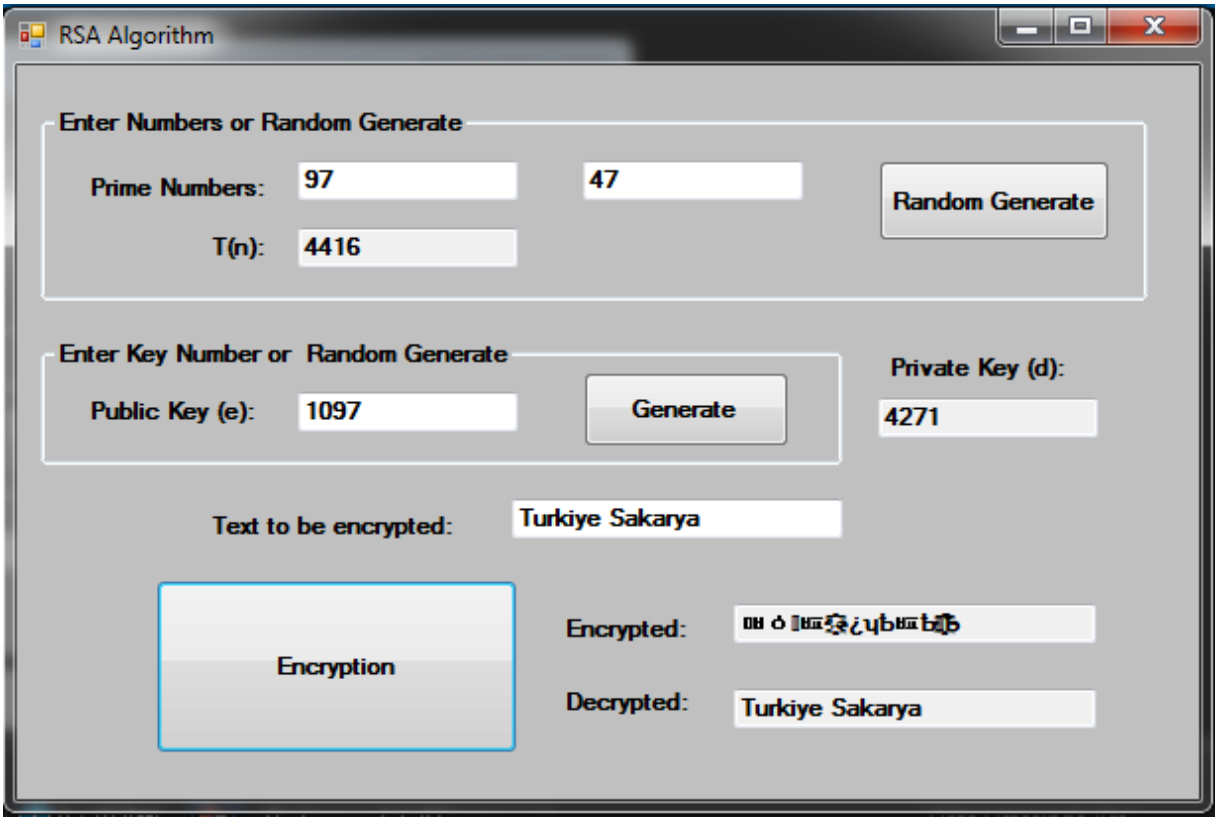
Şifrelenmiş metnin çözülmesi için, yine her bir karakterin ASCII değerine göre;

$$m = c^d \bmod(n)$$

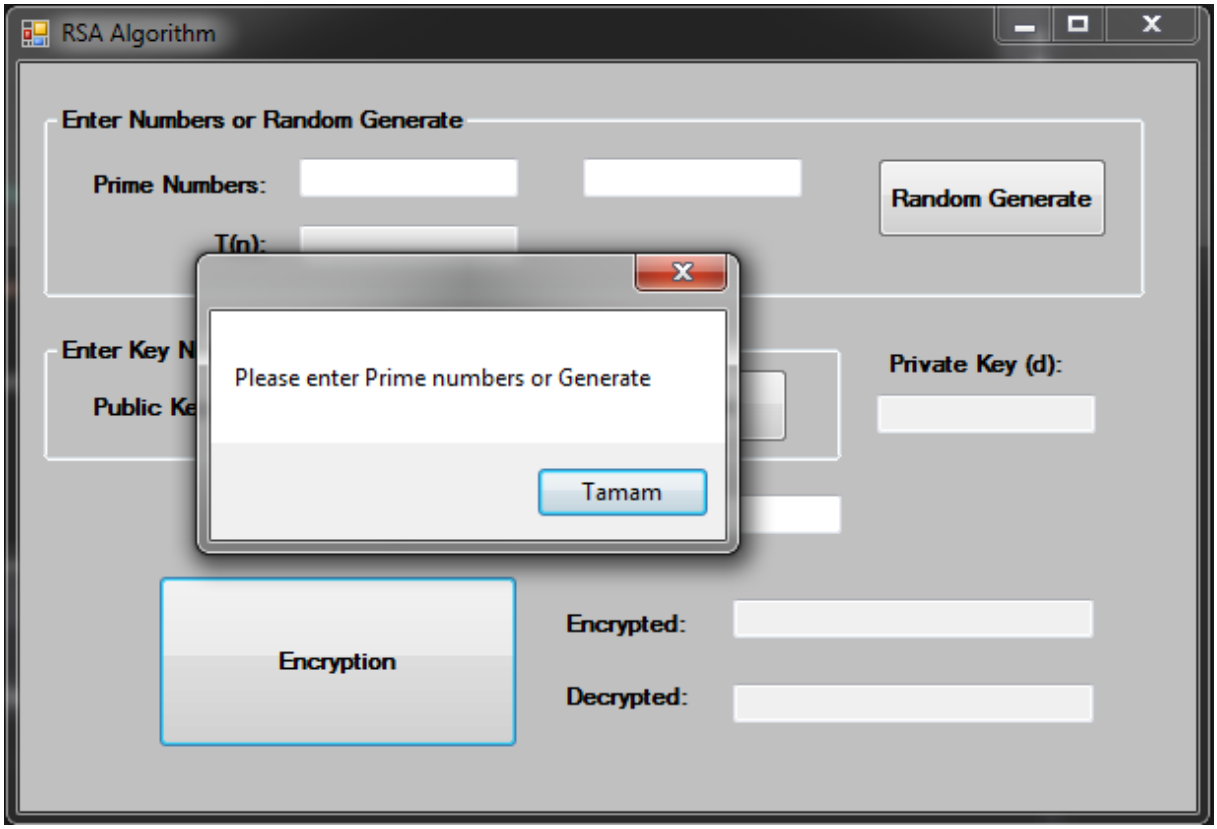
PROGRAM ÖRNEK EKRAN ÇIKTILARI

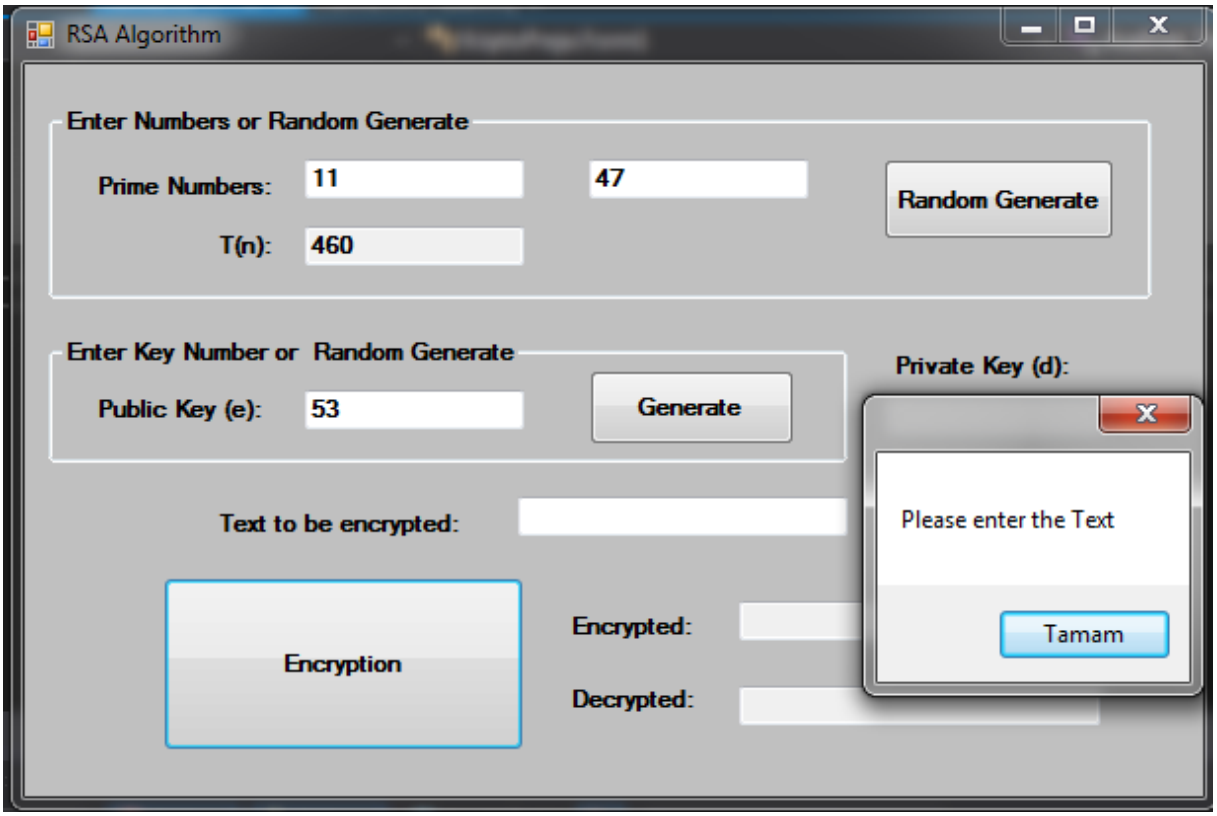
İlk grupta iki asal sayı girilmesi istenmektedir. Eğer kullanıcı sayı girmek istemezse butona basıp sayı üretmesi beklenmektedir. Bu iki durumun ortak sonucunda T(n) değeri hesaplanmış olacaktır. Hemen

altındaki ikinci bölümde ise açık anahtar olan e değerinin girilmesi veya Generate butonu ile rastgele üretilmesi beklenmektedir. En son şifrelenecek metin girilip Encrypted butonuna basıldığında, şifreleme ve gizli anahtar(d) ile doğrulama amaçlı şifreli metni çözme işlemleri yapılacaktır.



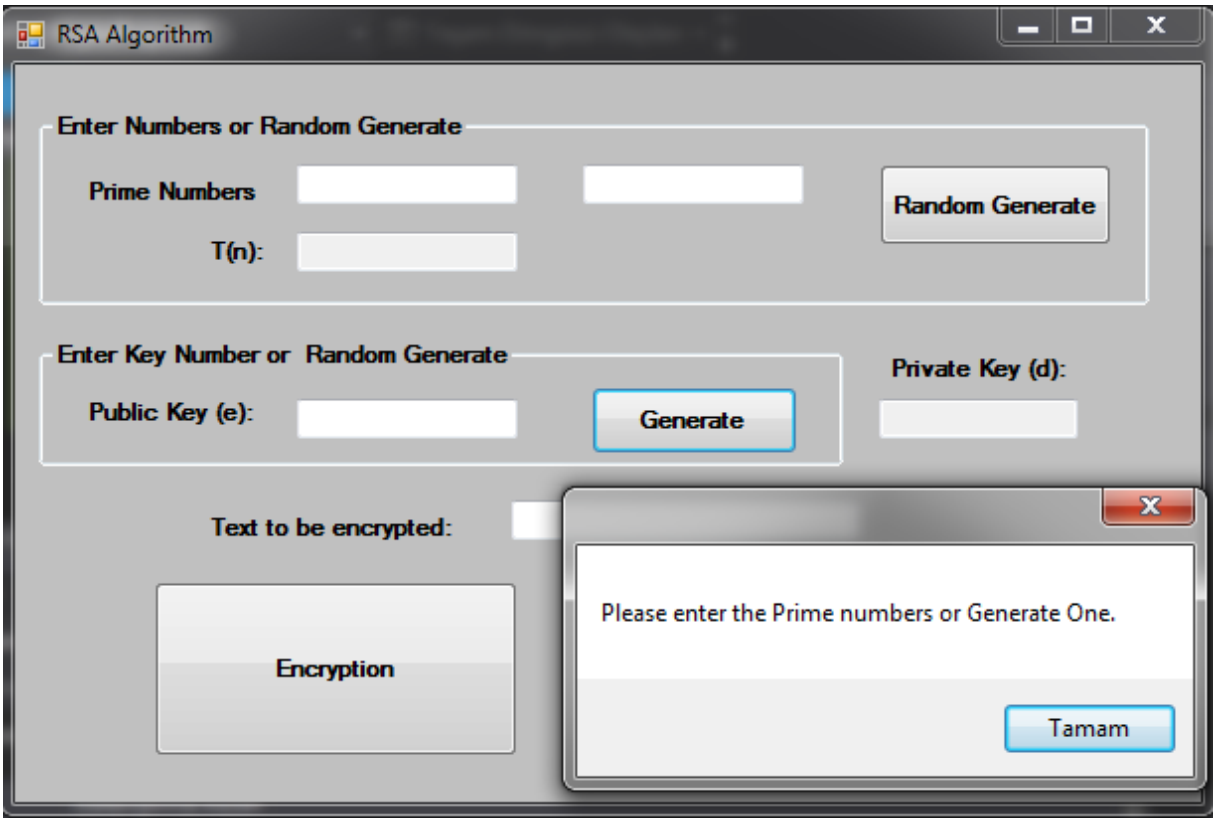
Eğer asal sayı girilmesi gereken yerler veya şifrelenecek metin girilmezse aşağıdaki uyarıları verir.



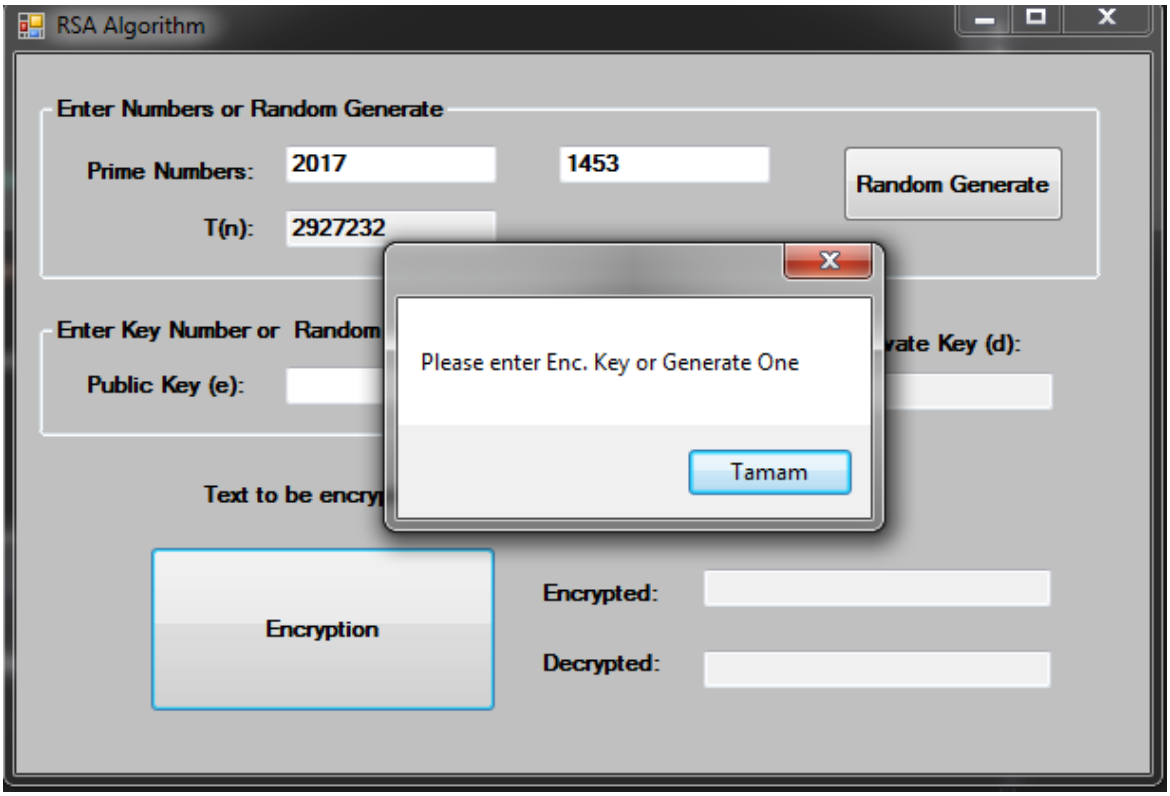


Diğer hatalar;

Eğer asal sayılar yokken e anahtarı üretilmek için Generate butonuna basılırsa, T(n) değeri ile aralarında asal kontrolü yapamayacağı için önce T(n) değerinin hesaplanmış olması beklenir. Bu değer olmadığı için hata verilmesi gerekir.



Açık anahtar(e) olmadan şifrele butonuna basılırsa yine hata verilir.



Eğer T(n) değeri ve açık anahtar varken, aralarında asal olup olmadığının kontrolü de yapılır. Değilse hata verilir.