Josipa Juričić

Split, 2019.

Šifriranje teksta

Plain Text to Cipher Text

Sadržaj

[Pretvaranje običnog teksta u šifrirani tekst 1](#_Toc13090509)

[1. Tehnika zamjene ( eng. Substitution Technique ) 1](#_Toc13090510)

[2. Tehnika prijenosa (eng. Transposition Technique ) 1](#_Toc13090511)

[Implementacija algoritama za šifriranje običnog teksta 2](#_Toc13090512)

[1. Keyword Cipher 2](#_Toc13090513)

[1.1. Implementacija u projektu 2](#_Toc13090514)

[2. Vigenere Cipher 5](#_Toc13090515)

[2.1. Implementacija u projektu 5](#_Toc13090516)

[3. Ceasar Cipher 7](#_Toc13090517)

[3.1. Implementacija u projektu 7](#_Toc13090518)

[4. XOR Cipher 8](#_Toc13090519)

[4.1. Implementacija u projektu 8](#_Toc13090520)

[5. Šifriranje lozinki kod kreiranja korisničkog računa 10](#_Toc13090521)

[5.1. Implementacija u projektu 10](#_Toc13090522)

# Pretvaranje običnog teksta u šifrirani tekst

Postoje dva osnovna načina na koji se obični tekst može modificirati kako bi se dobio šifrirani tekst: tehnika zamjene i tehnika prijenosa.

## Tehnika zamjene ( eng. Substitution Technique )

Tehnika zamjene uključuje zamjenu slova drugim slovima i simbolima. Na jednostavniji način zamjenjuju se znakovi običnog teksta, a umjesto njih se koriste drugi zamjenski znakovi, brojevi i simboli. Vrste tehnike zamjene:

* Caesar Cipher
* Mono Alphabetic Cipher
* Homophonic Substitution Cipher
* Polygram Substitution Cipher
* Vigenere Cipher
* Keyword Cipher
* XOR Cipher

## Tehnika prijenosa (eng. Transposition Technique )

U tehnici prijenosa identitet znakova ostaje nepromijenjen, ali se njihovi položaji mijenjaju kako bi se stvorio šifrirani tekst. Vrste tehnike prijenosa:

* Rail Fence Technique
* Simple Columnar Transposition Technique
* Vernam Cipher

U ovom projektu na jednostavan način je prikazana implementacija tehnika zamjene: Keyword Cipher, Caesare Cipher, Vigenere Cipher i XOR Cipher. Također je prikazan primjer upotrebe ugrađenih klasa i metoda korištenjem System.Security.Cryptography prilikom šifriranja korisničkih lozinki.

# Implementacija algoritama za šifriranje običnog teksta

## Keyword Cipher

Ovo šifriranje je oblik monoalfabetske supstitucije. Ključna riječ se koristi kao ključ i određuje podudaranja slova šifrirane abecede s običnom abecedom. Uklanjaju se ponavljanja slova u riječi, zatim se šifrirajuća abeceda generira s podudaranjem ključne riječi s A, B, C itd. Sve dok se ne iskoristi ključna riječ, nakon čega se ostala šifrirana slova koriste abecednim redom, isključujući slova već u ključu.

Abeceda: A B C D E G H I J K L M N O P R S T U V W X Y Z  
Šifrirano: K R Y P T O S A B C D E G H I J L M N Q U V W X Z  
Uz KRYPTOS kao ključnu riječ, svako A postaje K, svako B postaje R i tako dalje.

Poruka: "KNOWLEDGE"  
Kodirana poruka: IlmWjbaEb

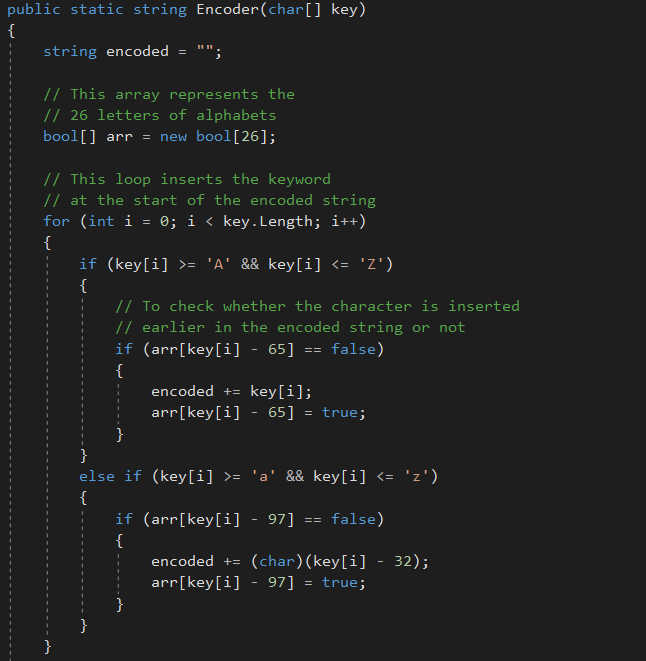
Da biste dekodirali poruku, provjerite položaj određene poruke kodiranjem teksta s običnim tekstom. Kako generiramo dešifrirani niz? Tražimo "P" u šifriranom tekstu i uspoređujemo njegovu poziciju s tekstom običnog teksta i generiramo to slovo. Dakle, "P" postaje "D", "T" postaje "E", "Y" postaje "C" i tako dalje.

* Sve su poruke kodirane velikim slovima.
* Razmak, posebni znakovi i brojevi ne uzimaju se u obzir u ključnoj riječi iako ih možete staviti tamo.
* Tijekom šifriranja poruke, razmaka, posebni znakovi i brojevi ostaju nepromijenjeni.

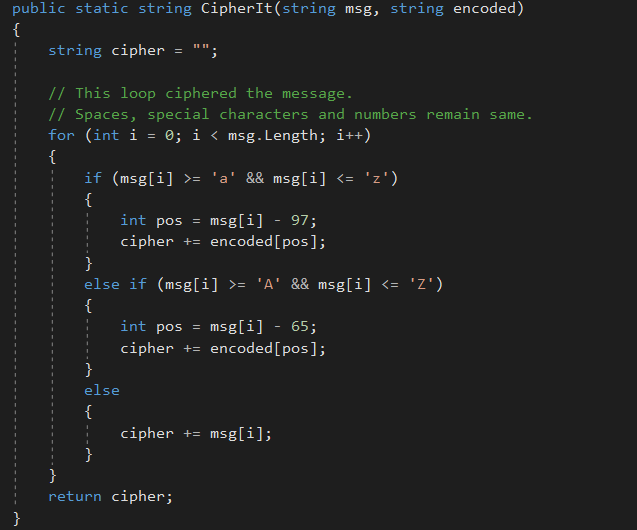
### Implementacija u projektu

Napravljena je statička klasa KeywordCipher koja sadrži tri statičke metode: Encoder, CipherIt i DecipherIt.

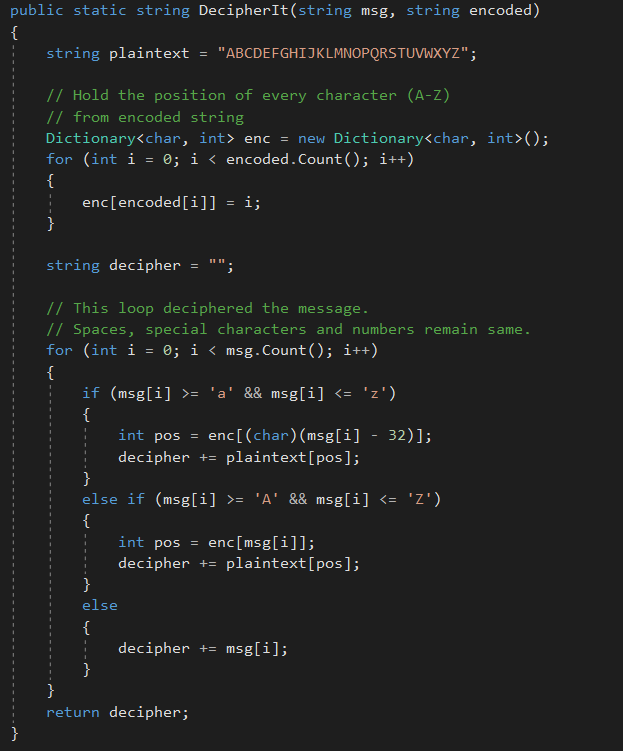
Statička metoda Encoder prima kao parametar ključ koji je niz znakova (char[] key ). Ova metoda uzima ključ koji stavlja na početak abecede ( ako ima slova koja se ponavljaju onda ih ne zapisuje dvaput) te nadopuni abecedu preostalim slovima iz abecede koja nisu iz ključne riječi. Na primjer, ključna riječ je „GAME“, onda je rezultat ove metode: „GAMEBCDFHIJKLNOPQRSTUVWXYZ“.

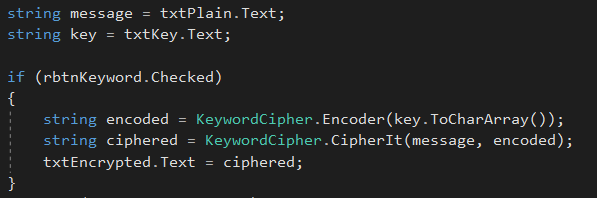


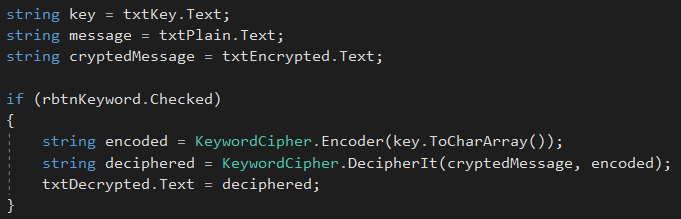
Statička metoda CipherIt prima kao parametre poruku za šifriranje i kodiranu abecedu ( string msg, string encoded). Ova metoda zamjenjuje svako slovo poruke pripadajućim slovom iz kodirane abecede. Pri tome, razmaci, posebni znakovi i brojevi ostaju isti.



Statička metoda DecipherIt prima kao parametre poruku za dešifriranje te kodiranu abecedu. Ova metoda zamjenjuje svako slovo šifrirane poruke pripadajućim slovom iste pozicije u standardnoj abecedi.







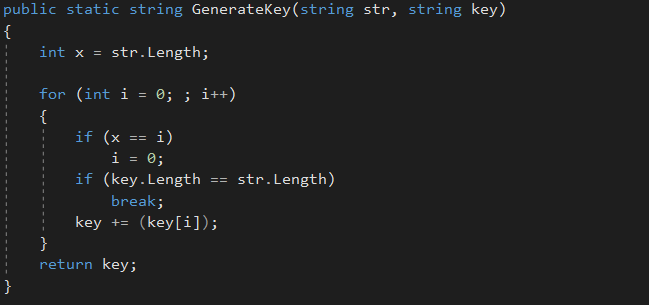
## Vigenere Cipher

Ova metoda je metoda šifriranja koja koristi jednostavan algoritam polialfabetske zamjene. Polialfabetska šifra je svaka šifra temeljena na zamjeni, a koja koristi višestruke zamjenske abecede. Šifriranje izvornog teksta se vrši korištenjem Vigenere-ovog kvadrata ili Vigenere-ove tablice. Tablica se sastoji od slova abecede zapisanih 26 puta u različitim redovima, svaka abeceda se ciklički pomiče ulijevo obzirom na prethodnu abecedu, što odgovara 26 mogućih Caesar šifri. Lakša implementacija bi bila vizualizirati Vigenère algebarski pretvaranjem [A-Z] u brojeve [0–25].

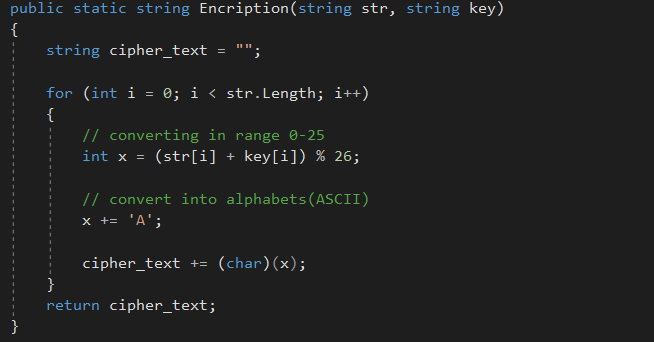
### Implementacija u projektu

Napravljena je statička klasa VigenereCipher koja sadrži tri statičke metode GenerateKey, Encription i Decryption.

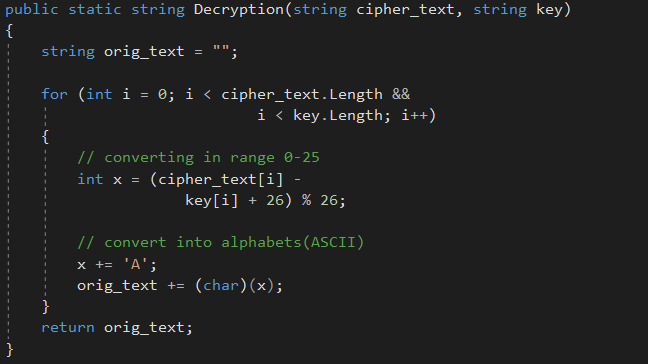
Statička metoda GenerateKey koja kao parametre prima početni tekst i ključ. Ova metoda generira ključ koji je iste duljine kao i početni tekst tako da ciklički ponavlja ključ sve dok nije iste duljine kao i tekst. Vraća generirani ključ iste duljine kao i tekst.

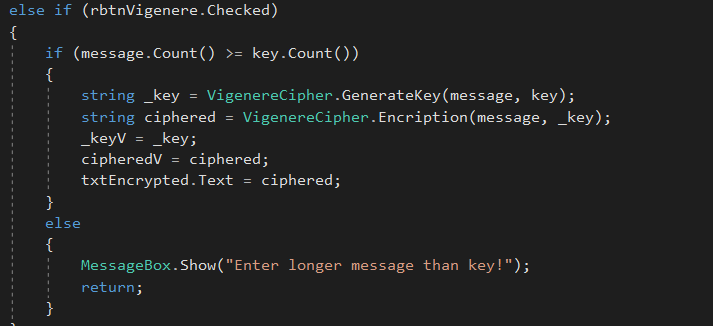


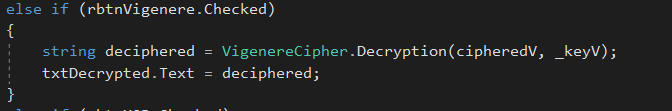
Statička metoda Encription kao parametre prima početni tekst i generirani ključ. Ova metoda svaki znak zamjenjuje brojem od 0 do 25. Za šifriranje koristi se formula: X = (M + K) % 26, gdje je M = pojedino slovo početnog teksta, K = pojedini znak ključa.



Statička metoda Decryption prima kao parametre šifrirani tekst i generirani ključ te vraća dešifriranu poruku. Za dešifriranje se koristi formula: X = (M – K + 26) % 26, gdje je M = pojedino slovo šifriranog teksta, K = pojedini znak generiranog ključa.







## Ceasar Cipher

Ova je metoda jedna od najstarijih i najjednostavnijih metoda šifriranja. To je jednostavan primjer tehnike zamjene, svako slovo teksta se zamjenjuje slovom s pozicije udaljene fiksni broj niz abecedu. Na primjer, s pomakom od 1, A se zamjenjuje s B, B sa C itd. Da bismo šifrirali dani tekst potrebna nam je cjelobrojna vrijednost, tj. pomak koji označava broj položaja za koji je svako slovo pomaknuto. Algoritam :

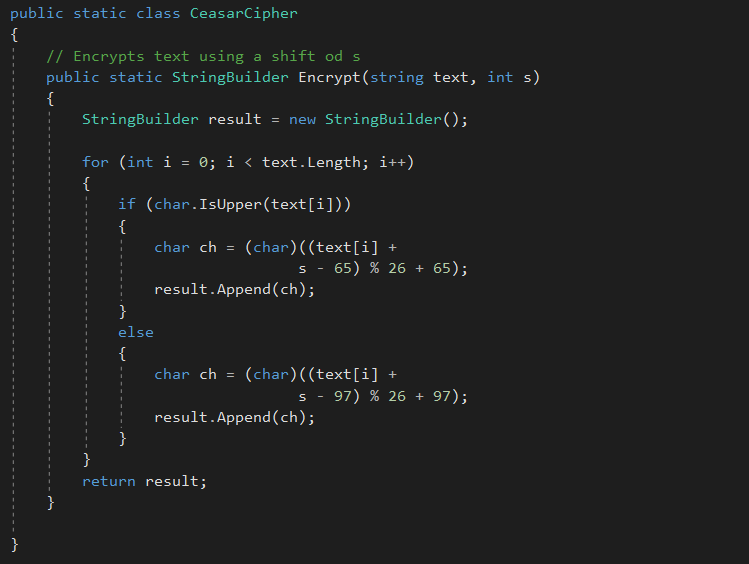
Ulaz: Niz malih slova. Cijeli broj između 0-25, označava potrebni pomak.

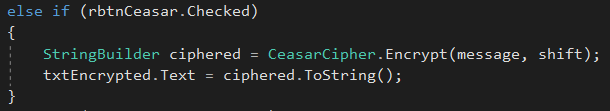
Postupak: Prolaziti zadani tekst po jedan znak. Za svaki znak transformirajte dati znak prema pravilu, ovisno o tome šifriramo li ili dešifriramo tekst. Vrati novi generirani niz.

Za dešifriranje možemo ili napisati još jednu funkciju šifriranja, koja će primijeniti danu promjenu u suprotnom smjeru za dešifriranje izvornog teksta, ili jednostavno koristiti istu funkciju, a promijeniti pomak tako da je pomak = 26 – pomak.

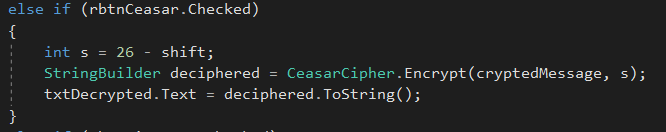
### Implementacija u projektu

Statička klasa CeasarCipher sadrži jednu statičku metodu Encrypt koja služi i za šifriranje i dešifriranje. Metoda prima kao parametre početni tekst i cjelobrojnu vrijednost pomaka. Ova metoda koristi formulu za pomicanje svakog slova abecede za isti pomak: X = ( M + S – 65(ili 97 za mala slova)) %26 + 65(ili 97 za mala slova).





Kod dešifriranja potrebno je samo metodi kao pomak poslati S = 26 – S.



## XOR Cipher

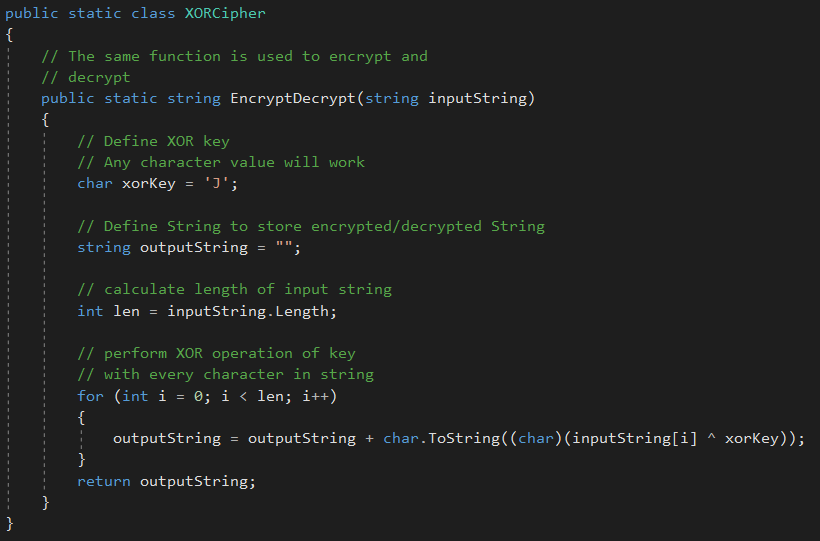
XOR šifriranje je način šifriranja kojim je teško probiti šifru Brute Force metodom tj. generirajući slučajne ključeve i provjeravati podudaraju li se s ispravnim ključem za dešifriranje. Koncept implementacije je prvo definirati XOR ključ za šifriranje, a zatim izvršiti XOR operaciju svakog znaka u tekstu sa ključem. Za dešifriranje šifriranih znakova moramo ponoviti XOR operaciju sa definiranim ključem.

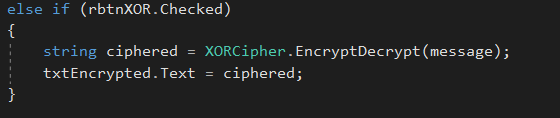
Osnovna ideja iza XOR šifriranja je da ako ne znate XOR ključ za šifriranje nemoguće je dešifrirati podatke. Na primjer, ako radite XOR operaciju nad dvije nepoznate varijable nemoguće je znati koji je rezultat. Ako imamo A XOR B, a rezultat je 1 tj. istina. Ako znamo jednu od varijabli onda znamo i drugu. Ako je A istina onda B treba biti laž ili ako je A laž onda B treba biti istina, prema tablici XOR operacije. Bez da znamo vrijednost jedne od varijabli ne možemo dešifrirati podatke.

### Implementacija u projektu

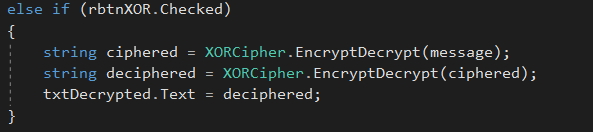
Korištena je statička klasa XORCipher koja sadrži jednu metodu za šifriranje i dešifriranje.

Statička metoda EncryptDecrypt koja prima kao parametar početni tekst vraća šifrirani/dešifrirani tekst. Metoda koristi XOR ključ koji može biti bilo koji znak. Metoda obavlja XOR operaciju nad svakim znakom iz početnog teksta i XOR ključa.



Na Windows Formi klikom na botun Encrypt u varijablu ciphered sprema se šifrirani tekst pozivanjem metode kojoj smo poslali tekst iz txtPlain textBox-a te se ispisuje na txtEncrypted textBox. 

Klikom na botun Decrypt ispisuje se na txtDecrypted rezultat iste metode kojoj smo sada poslali šifrirani tekst.



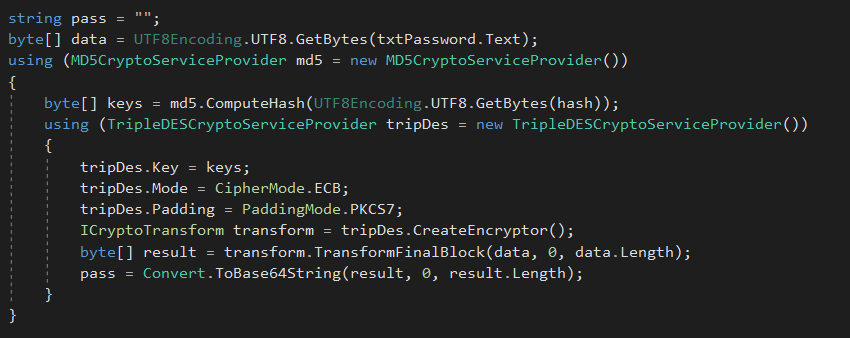
## Šifriranje lozinki kod kreiranja korisničkog računa

Jedan od primjera gdje se može koristiti šifriranje je kod spremanja podataka korisničkog računa u bazu podataka kojoj mogu na primjer pristupiti mnogi zaposlenici firme. U ovom projektu korišten je jednostavan način spremanja podataka u jednu tekstualnu datoteku. Prilikom spremanja podataka o korisniku u datoteku ne sprema se lozinka koju korisnik unese već njena šifrirana verzija. Prilikom korisnikova uvida u korisnički profil ( Log in ) dešifrira se lozinka pomoću ključne riječi kojom je i šifrirana. Na ovaj način omogućeno je da se bez poznavanja ključa za šifriranje ne može pristupiti podacima o lozinkama korisničkih profila.

### Implementacija u projektu

Ovdje je korišteno using System.Security.Cryptography, ugrađeno okruženje sa svim potrebnim klasama i metodama za enkripciju podataka.

Prilikom spremanja podataka u tekstualnu datoteku lozinku je prije potrebno šifrirati.



Prilikom čitanja iz datoteke lozinka se dešifrira istim ključem (string hash) kojim je i šifrirana.

