VERİ SIKIŞTIRMA PROJESİ

Osman Şimşek
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
180202048
simsekosman15@hotmail.com

Yener Emin Elibol
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
180202054
yenereminelibol@gmail.com

Veri Sıkıştırma Projesi karakter tabanlı sıkıştırma algoritması kullanarak kullanıcının girdiği verileri sıkıştıramayı sağlayan projedir.

I. GİRİŞ

Veri Sıkıştırma Projesi karakter tabanlı sıkıştırma algoritması olan lz77 algoritmasını kullanarak kullanıcının proje dosyalarının konumunda bulunan "Metin.txt" dosyasına girdiği verileri sıkıştırır. Lz77 algortimasında ki temel amaç metin içinde tekrar eden kısımları tespit edip token denilen yapılar kullanılarak tekrar etmeyen ve tekrar eden kısımlara tokenler koyup sıkıştırılma yapması beklenmektedir. Tokenlerin temel amaçlarından biride sıkıştırma yapılan dosyanın eski haline getirilmesi için kolaylık sağlamaktır.

II. TEMEL BİLGİLER

.Program C programlama dilinde gerçekleştirilmiş olup geliştirme ortamı olarak CodeBlocks kullanılmıştır.

III. TASARIM

A. Algoritma

Kullanıcı sisteme veri girişi yapıp program çalıştırıldığında sistem kullanıcının girdiği verileri Metin.txt dosyasından sisteme çekerek lz77 algoritması uygulamak üzere fonksiyona göndermektedir. LZ77 algoritmasının amacı, metin belgelerindeki tekrar eden bölümleri ortadan kaldırmak ve buna yönelik metin belgesini tokenler yardımıyla sıkıştırmaktır. Lz77 metnin içerisinde LZ77 yaklaşımında sözlük, daha önce kodlanmış serinin bir parçasıdır. Algoritmadaki arama tamponunun büyüklüğü, daha önce kodlanmış serinin ne büyüklükte bir parçasında arama yapılacağını belirler. Arama tamponu büyütüldükçe, sıkıştırma oranı artar, fakat aynı zamanda sıkıştırma zamanı da artar. Arama tamponu bu programda en fazla 4096 karakter öncesinden arama yapmaya başlayabilmektedir ve ileri tampon miktarı ise en fazla 16 karakter uzunluğundadır.

Arama tamponu ile ileri tampon arasında benzer bir kelime bulunursa bu benzerliğin ne kadar uzunlukta olduğu bulunur. Bulunan benzerliğin ana konumdan ne kadar gerisinde başladığı bilgileri ileri tampon ve arama tamponu yardımı ile bulunur. Bulunan bu değerleri token oluşturulup içerisinde bulunan 16 bit boyutundaki bir değişkene aktarılmaktadır. Bunlardan 12 si arama tamponuna 4 ü ise ileri tampon bilgisini tutmaktadır. Eğerki arama ve ileri tampondan gelen değerleri int tipinde tutsaydık tokenimizin boyutu artıcaktı ve dosyamızın boyutuda artıcaktı bunun yerinde 16 bitlik bir değişken kullanarak sıkıştırma yapılmaktadır. Tokenlerimizin içerisinde eşleşmeden sonra gelen ilk değer tutulmaktadır. [10,7,a] Tokenlerin genel gösterimleri bu şekildedir ilk değer arama tampon bilgisi ikinci değer ileri tampon bilgisi 3. değer eşleştirmeden sonra gelen ilk karakterdir eğerki eşleştirme bulunamadıysa o anki karakterin tokeni [0,0,a] gibidir. Her adımda token oluşturulmaktadır eşleşme yakalansa da yakalanmasa da. Eğerki hiçbir şekilde sıkıştırılamadıysa dosya normalde karakterin boyutu 1 bytetır fakat Tokenlerin boyutu karakterin boyutunun 3 katına sahip olduğundan dosya boyutu 3 katına çıkmaktadır. Lz77 algoritması kısa metinlerde veya eşleşme olmayan metinlerde işe yaramamaktadır.

B. Kullanılan Bazı Fonksiyonlar

- struct token *lz77encode:
  Programın ana fonksiyonlarından birisidir. Metin.txt içerisinden okunan veriyi parameter olarak alıp token yapıları ile sıkıştırma işlemini gerçekleştirmektedir. Sıkıştırma işlemi gerçekleştirildikten sonra oluşturulan tokenleri return etmektedir.

- char offsetLenghtOlustur:
  Lz77 yapısında arama tamponu ve ileri tampon olarak iki yapından oluşmaktadır. Arama tamponu ve ileri tampondan gelen verileri token içerisinde yeralan 16 bitlik değişkenin içine 12 biti arama tamponu 4 biti ileri tampon olacak şekilde yerleştirmektedir.

- char* fileRead:
  Bu fonksiyon kullanıcının Metin.txt içerisine girdiği verileri okuyarak lz77 nin üzerinde işlem yapması için kaydetmektedir.

C. Karşılaşılan Bazı Sorunlar

- Bu projede karşılaşılan sorunlardan biri token yapısını oluşturmak, arama tamponu ve ileri tampondan gelen bilgileri 16 bit boyutundaki değişkenin içerisine yerleştirmek oldu.
- Lz77 algoritmasının çalışma mantığını anlamakta bazı sorunlar yaşadık fakat kaynakçada belirtilen yerlerde sorunumuzu çözecek bilgiler bulduk.

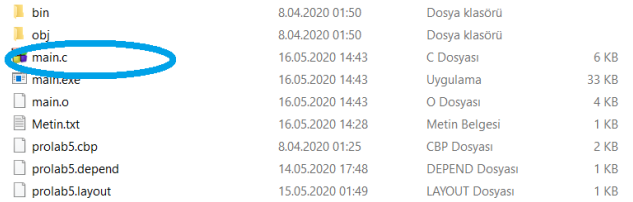
D. Kazanımlar

- C de pointer yapısının kullanımını daha iyi anladık.
- C de bit düzeyinde işlem yapmayı ve stdint.h kütüphanesinden gelen değişkenleri ile çalışmayı öğrendik.
- Lz77 algoritmasının nasıl bir yöntem ile sıkıştırma yaptığını öğrendik.

E. Çalışma Adımları

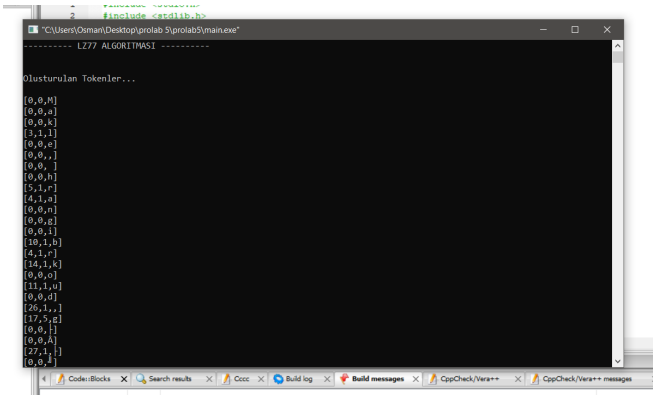
Adım 1:

Proje veri sıkıştırma projesi olduğundan dolayı bir metin dosyasına girilen yazıların sıkıştırılması beklenmektedir. Bu yüzden sistem dosyalarının olduğu yerdeki 'Metin.txt' dosyasının içerisine sıkıştırmak istediğiniz metni girmeniz gerekmektedir.



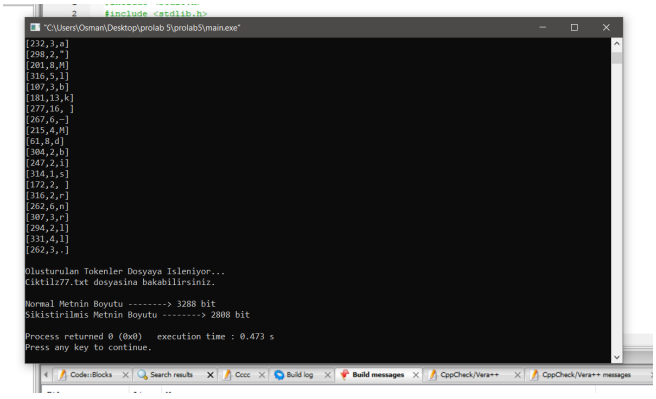
Adım 2:

Metin.txt dosyasının içerisine verileri girdikten sonra program çalıştırılmak istendiğinde main.c dosyasına girerek 'F9' tuşuna basarak codeblock aracılığı ile çalıştırabilirsiniz ya da main.c dosyasının hemen altında yeralan main.exe ye tıklayarak programın direk çalışması sağlanabilir.



Adım 3:

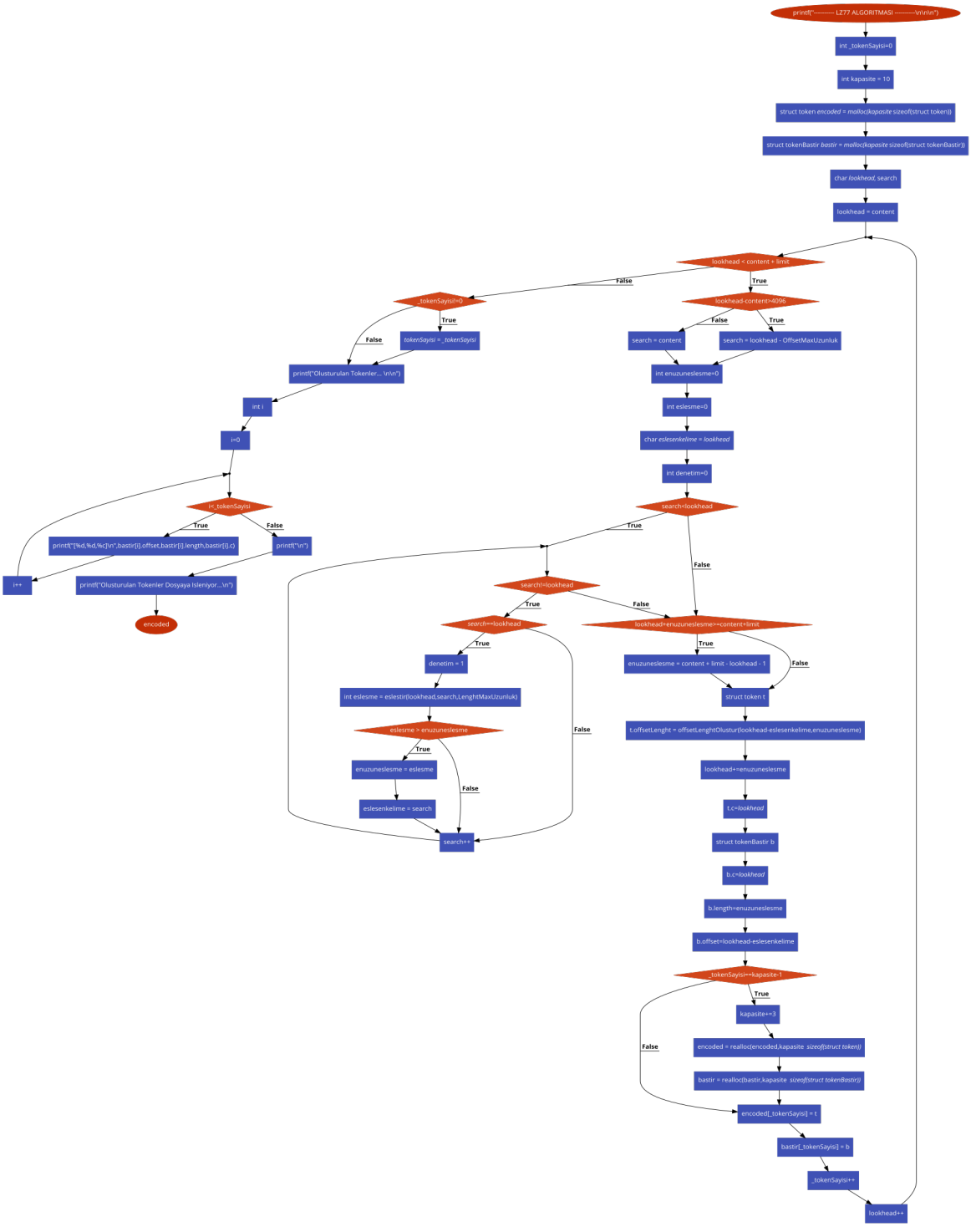
Yandaki resimde de görüldüğü üzere program çalıştırıldığında karşınıza cmd ekranı gelmektedir. Bu ekranda dosyanın sıkıştırıldığı bilgisi verilmektedir. Lz77 ye göre sıkıştırılan verilerden oluşan tokenler ekrana basılmaktadır.



Adım 4:

Dosya sıkıştırılması tamamlandığı vakit cmde ekranının en aşağıya doğru indiğinizde sıkıştırma sonucu oluşan tüm tokenler gösterilmekte. En aşağıda sıkıştırmanın bittiği bilgisi ekranda gösterilmektedir. Sıkıştırma tamamlandığında normal metnin boyutu ile sıkıştırılan metnin boyutu ekrana basılmaktadır. Sıkıştırma tamamlandıktan sonra ototmatik olarak proje dosyalarının yer aldığı yere 'Ciktilz77.txt' dosyası oluşturularak sıkıştırılan metin içerisine basılmaktadır. Projeye sonlanınca cmd ekranında sonda kullanıcıdan bir girdi beklemektedir. Bu cmd ekranının kapanmaması için alınan bir önlemdir. Projeyi sonlandırdıktan sonra 'Ciktilz77.txt' dosyasını kontrol edebilirsiniz.

F. Akış Diyagramı



lz77encoded fonksiyonunun çalışma sistemi

KAYNAKÇA

[1] http://altanmesut.trakya.edu.tr/pubs/dr_tez.pdf

[2] https://tr.qwe.wiki/wiki/LZ77_and_LZ78

[3] https://www.youtube.com/watch?v=FvWQsuwbraA

[4] http://ysar.net/algoritma/lz77.html

[5] https://www.slideshare.net/veysiertekin/lz77-lempelziv-algorithm