

Problem: Yazacağınız program C++ konsol uygulaması olup çalıştırıldığında Sayilar.txt isimli dosyayı okuyacaktır. Bu dosya içerisinde her satırda 0-255 arası pozitif sayılar bulunacak ve sayıları birer boşluk ayıracaktır. Dosyadaki her satır için bir adet iki yönlü dairesel bağlı liste nesnesi oluşturulacaktır. Sırayla sayılar okunup aşağıdaki algoritmaya göre listelere eklenecektir.

Belirlenen problem doğrultusunda belirli matematiksel hesaplamaların yapılması ve girdi olarak alınan verinin işlenip bilgi elde edilmesi amaçlanmıştır.

Bu doğrultuda file read, struct, circle linked list, array, memory allocated gibi yapılar projede kullanılmıştır.

Dosyanın okunması için dosyanın adını parametre olarak alan bir fonksiyon hazırlanmıştır. Bu fonksiyon dosya okuma işlemlerini FILE struct yapısından bir pointer üreterek okumaktadır.






Ad	Değiştirme tarihi	Tür	Boyut
bin	15.08.2020 01:02	Dosya klasörü	
doc	15.08.2020 10:07	Dosya klasörü	
include	15.08.2020 00:51	Dosya klasörü	
lib	15.08.2020 01:01	Dosya klasörü	
src	15.08.2020 01:02	Dosya klasörü	
benioku.txt	15.08.2020 01:02	Metin Belgesi	1 KB
makefile	15.08.2020 01:02	Dosya	1 KB
Sayilar.txt	15.08.2020 10:08	Metin Belgesi	1 KB

Sayilar.txt dosyası ana dizinde bulunmaktadır. Makefile dosyası ile derleme sonucu makefile dizininde çalıştırılmaktadır program bu nedenle veri dosyası programın çalıştığı dizinde bulunmalı veya tam yolu girilmelidir.

bin/main.exe ana çalıştırılabilir programdır ve bulunduğu dizinde bir terminal açılıp çalıştırıldığında eğer aynı dizinde veri dosyası yok ise program hata verecektir. Bu nedenle doğrudan .exe çalıştırılmak isteniyorsa bulunduğu dizine veri dosyası(Sayilar.txt) biçiminde eklenmelidir.

Veri dosyasının sonunda bir satır boşluk olmalı ve sayılar birbirinden boşluk karakteri ile ayrılmalıdır.

Sayilar.txt - Not Defteri					
Dosya	Düzen	Biçim	Görünüm	Yardım	
82	75	65	83	65	121 65

Ad	Değiştirme tarihi	Tür	Boyut
 fileOperation.hpp	15.08.2020 01:01	C++ Header File	2 KB
 includeOperation.hpp	15.08.2020 00:58	C++ Header File	1 KB
 listOperation.hpp	15.08.2020 00:53	C++ Header File	2 KB
 numOperation.hpp	15.08.2020 00:54	C++ Header File	1 KB
 printOperation.hpp	15.08.2020 01:00	C++ Header File	1 KB

İlgili işlemler farklı başlık dosyalarına ayrılmış olup includeOperation.hpp dosyası içinde hiyerarşik bir biçimde dahil edilmiştir.

Program temel olarak basit bir şifreleme işlemi yapmaktadır. Şifreleme için temel olarak mod alma işlemlerini kullanmaktadır.

Şifreleme işleminde ASCII tablosu kullanılmıştır. 0-255 arasında sembol, rakam ve harf gibi değerler barındırmaktadır.

Bu proje doğrultusunda şifreleme işlemlerinin matematiksel işlemler üzerinden yapıldığı üretimi kolay, geri dönüşü zor olan mod alma gibi işlemler ile yapıldığı öğrenilmiştir.

Belirlenen algoritma her girdi için dinamik olarak o girdinin değerine göre çalışmaktadır.

Veri dosyası içerisindeki her satır ayrı olarak değerlendirilir.

Her satırdaki her bir sayı ayrı olarak değerlendirilir.

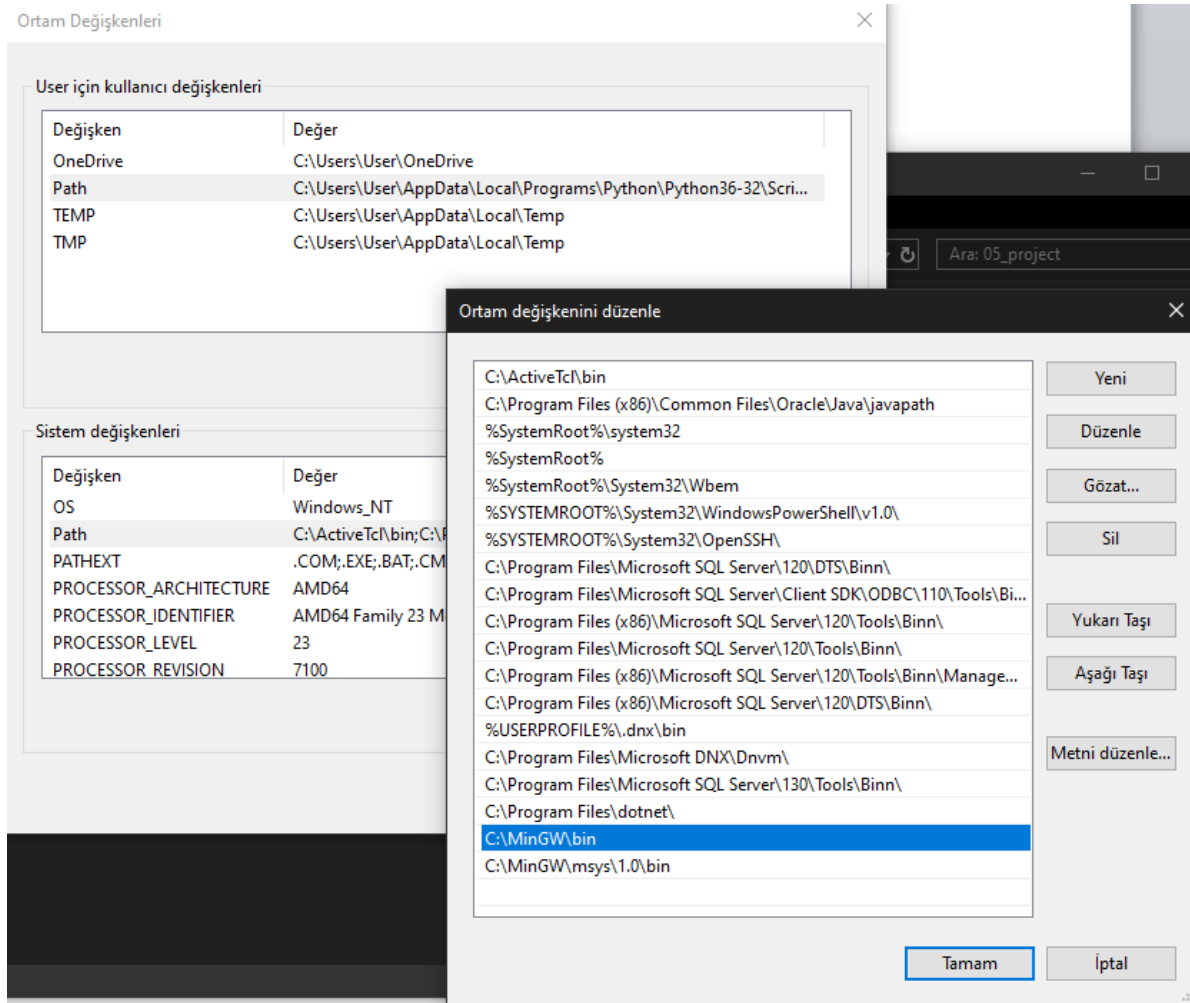
O anki satırdaki her bir sayı ilgili algorithmadan geçirilip gerekli işlemler yapıp listedeki konumu belirlenip, ardından listedeki ilgili alana yerleştirilir.

Bu işlemler sırasında kaydırma işlemi yapılarak veriler şifrelenmiştir.

Windows işletim sistemleri için hazırlanmış olan Mingw kullanılmıştır.

Mingw içerisinde g++ derleyicisi kullanılarak nesne dosyaları ve çalıştırılabilir dosya üretilmiştir.

Dikkat edilmesi gereken;



Bilgisayarın Ortam değişkenleri bölümünden MinGW\bin ve MinGW\msys\1.0\bin yollarının tanımlı olmasıdır.

Bu işlemle birlikte terminal herhangi bir yerde açıldığında C:\MinGW\bin dizinindeki tüm programlar kullanılabilir olacaktır.

Böylelikle proje dizininde “make” komutu kullanılarak proje derlenebilir.