



## **2. Odev RAPORU**

### **HAZIRLAYAN**

**ADI SOYADI** : M. FATİH KARATAŞ

**ÖĞRENCİ NUMARASI** : B211210088

**TESLİM TARİHİ** : 24/12/24

**DERS ADI** : Veri Yapıları

**ÖĞRETMEN ADI** : Prof.Dr. M. Fatih Adak

**DERS GRUBU** : A

**Ödev Numarası** : 2

**Ödev KONUSU** : BST - Ağaç çizdirme

## 1. İSTENİLENLER

- \* Bağlı liste ile düğüm yönetimi
- \* Her düğümde BST ağaç yapısı
- \* ASCII değerlerine göre hesaplama
- \* Görsel arayüz ve kontroller
- \* Dosyadan veri okuma

## 2. ÖĞRENDİKLERİM

- \* Veri yapılarının birlikte kullanımı (LinkedList + BST)
- \* Rekürsif fonksiyonların önemi
- \* ASCII değer hesaplamaları
- \* Bellek yönetimi (destructor kullanımı)
- \* OOP prensipleri
- \* Template programlama (Vector sınıfı)

## 3. PROJEDE YAPTIKLARIM

- \* Dosyadan satır satır okuma ve işleme
- \* BST ağaç yapısının oluşturulması
- \* ASCII değerlerinin hesaplanması
  - Sol düğümler \* 2
  - Sağ düğümler normal toplam
  - Kök değeri ekleme
- \* Görsel arayüz ve navigasyon
- \* Aynalama operasyonu
- \* Düğüm silme işlemleri

## 4. EKSİK BIRAKTIĞIM YERLER

- \* BST dengeleme yapılmadı
- \* Hata kontrolü geliştirilebilir
- \* Bellek optimizasyonu yapılabilir
- \* Kod tekrarları azaltılabilir

## 5. ZORLANDIĞIM KISIMLAR

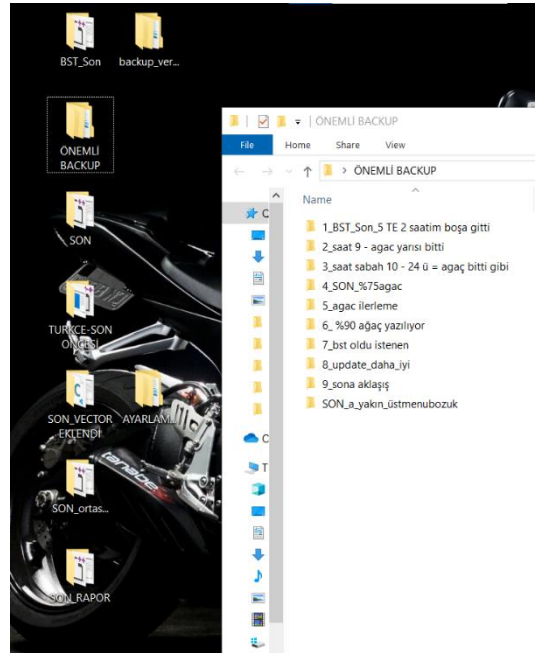
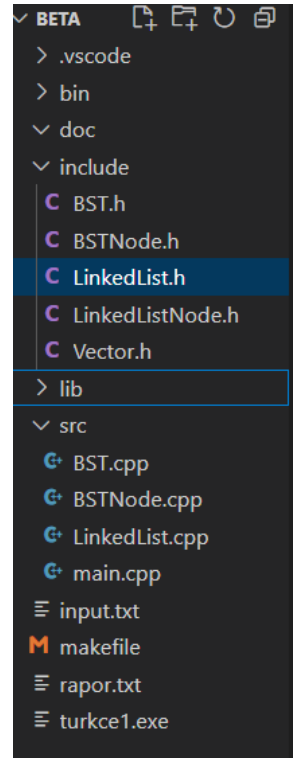
- \* ASCII hesaplama mantığının oturtulması
- \* BST ağaç görselleştirilmesi
- \* Aynalama sonrası ASCII değer güncelleme
- \* Bellek yönetimi
- \* Pointer işlemleri

## 6. ÇÖZÜM YAKLAŞIMLARIM

- \* Adım adım geliştirme
- \* Test güdümlü yaklaşım
- \* Modüler kod yapısı
- \* Sınıf hiyerarşisi
- \* Fonksiyon bölümleme

## 7. SON DURUM

- \* Temel işlevler çalışıyor
- \* ASCII hesaplamaları doğru
- \* Görsel gösterim başarılı
- \* Kullanıcı kontrolü sağlandı
- \* Bellek sızıntısı önlendi



= Odev kayıt aşamalarım