

T.C.  
SAKARYA ÜNİVERSİTESİ  
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ  
VERİ YAPILARI DERSİ  
1. ÖDEV

DERS ADI: VERİ YAPILARI

DERSİ VEREN ÖĞRETİM ÜYESİ: MUHAMMED FATİH ADAK

ÖĞRENCİ ADI: ALİ HALİT ÖRNEK

ÖĞRENCİ NUMARASI: B231210385

DERS GRUBU: 1 A



**SAKARYA**  
ÜNİVERSİTESİ

## ÖDEV İNCELEMESİ

Kodun genel amacı, dairesel bağlı liste veri yapısını kullanarak kromozomlardan oluşan bir dna'da çeşitli manipülasyonlar gerçekleştirmek. Arayüzde bir menü vardır. Menüde 5 farklı işlem gerçekleştirilebilir bunlar sırasıyla çaprazlama, mutasyon, otomatik işlemler, ekrana yaz ve çıkıştır. Ödevde 4 farklı sınıf kullanıldı. Bunlar Daireselliste, Dna, Dugum ve SatirDugum dür. Daireselliste, kodun ana veri yapısıdır ve düğümlerden oluşan kromozomları modellemek için kullanılır. Bunun yanında çaprazlama ve mutasyon gibi işlemler buradan yapılır. Dugum, Dairesellistedeki karakterleri birbirine bağlamak için kullanılır. Önceki, sonraki işaretçilerini ve veriyi kromozomun her bir harfi için tutar. SatirDugum, Her bir satırında bir Daireselliste tutar. SatirDugum, kromozomların satır bazında düzenlenmesini sağlamaktadır. Dna sınıfı ise dosyayı okumamızı ve kromozomlar üzerinde çeşitli işlemler ve manipülasyonlar için bir arayüz sağlar. Program çalıştığında ilk olarak Dna.txt okunur. Bu işlem sayesinde her bir satır yani kromozom bir dairesel liste olarak tutulur. Sonrasında gelen ekranda kullanıcı istenilen işlemi seçer. Çaprazlama işlemi 2 farklı kısımdan yani fonksiyondan oluşur. Bunlar birbirini tamamlayan 2 parçadır tek parça olamamasının sebebi ayrı ayrı oluşan kromozomların 2 seferde dna'ya eklenmesidir. Mutasyon işlemi ise istenilen satır ve istenilen sıradaki adresi bulur ve bu adresin verisini 'X' olarak değiştirir. Otomatik işlemler bize verilen İşlemler dosyasındaki işlemleri otomatik olarak uygulamamızı sağlar. Ekrana yaz ise sola doğru giderken ilk düğümün verisinden daha küçük bir veri varsa onu ekrana yazmamızı sağlar. Bu işlem tüm kromozomlar için uygulanır. Bu işlemi göz önünde bulundurduğum için dairesel olan ve önceki işaretçisi olan bir veri yapısının bu işi kolaylaştıracağını düşündüm bu yüzden ödevde iki yönlü dairesel bağlı liste kullandım. Çıkış işlemi ise programdan çıkış yapmamızı sağlar. Bu işlemleri yapmamızı sağlayan yardımcı fonksiyonlara bakarsak. Ekle listeye dugumu kullanarak veri ekler. SatirEkle ise listeleri satirDugum kullanarak birbirine ekler. DugumGetir istediğimiz sıradaki bir düğümü getirir. SatirGetir istediğimiz sıradaki listenin adresini getirir. ListeyiYazdır istenilen listeyi yazdırır. Bu fonksiyonu mutasyon işleminden sonra bir kez çağırıyorum ama aslında programı yazarken tüm kromozomları görmem gerekiyordu. İşlemlerin doğru gerçekleşip gerçekleşmediğini anlamak için daha çok orada kullandım. KromozomuGuncelle fonksiyonu ise mutasyon işleminden sonra listenin yeni halinin kaydedilmesini sağlar. Yıkıcı metotlarda ise ~Daireselliste metodu boş bırakıldı. Bunun sebebi dosyaOku fonksiyonu çağırıldıktan sonra oluşan listeler biz program içerisinde kullanmadan oluşup yıkılıyordu. Bu yüzden bu iş için ~Dna yıkıcı metodu kullanıldı. Bu metot içerisinde dairesel listeleri temizlemek için bir temizle fonksiyonu çağırıldı bu aslında bir çeşit ~Daireselliste yıkıcı fonksiyonu yerine tanımlanan fonksiyondur. Bu şekilde hem dugumlerin tuttuğu dairesel listeler silindi hemde satir dugumlerin tuttuğu dairesel liste listeleri temizlenmiş oldu. ~Dugum ve ~SatirDugum metotları da boş bırakıldı çünkü zaten tüm veri temizlendi bunların içerisinde silme yaparsak 2 defa silme hatası alırız.