

Öncelikle yığıt yapısını oluşturmak için Stack isimli class oluşturdum ve yığıtın gerekli görevlerini içine tanımladım. Dolayısıyla verileri tutmamız için bir yığıt yapımız oluşmuş oldu. Fakat bu verileri her düğümün kendine özel yığıtı olacak şekilde tutacağı için yeni bir Dugum class'ı oluşturdum. ve Stackin nesnesini pointer şeklinde bu class'ın içinde oluşturdum. Dugum class'ının içinde ayrıca 2 Dugum nesnesini pointer şeklinde oluşturdum. Bu nesneler Dugum tipli prev ve next. Bunları oluşturmamın sebebi düğümlerin arasında gezebilmek. Daha sonra çift yönlü dairesel bağlı listenin görevlerini yazabilmek için DaireselBagilListe adında class oluşturdum. Bu class'ın içinde pSimdiki adlı en baş düğümün adresini gösterecek bir dugum tipli pointer nesnesi tanımladım. Geriye çift yönlü dairesel bağlı listemize düğümlerimizi ekleyecek, bunların içinde stacklerimizi oluşturacak ve stacklerimizin içine verileri aktaracak komut olan Ekle(veri)'yi yazmak, SagaHareket()'i yazmak, SolaHareket()'i yazmak, bir düğümün durumunu yazacak olan komut DurumYaz(konum)'u yazmak ve tüm durumları yazacak olan komut TumDurumlariYaz()'ı yazmak kalıyor. Ekle(veri) komutunda sona eklenecek olan yeni düğüm oluşturunca pSimdiki'nin adresini alacak bir geçici düğüm oluşturunca. Geçici düğümün adresini eleman sayısı kadar ilerisindeki adrese atıyorum next düğümünü kullanarak. Artık geçici düğümümüz son elemanımıza geliyor. Bir ilerisine yeni düğümü atayacak işlemleri gerçekleştiriyorum ve içerisindeki yığıta parametre olarak aldığımız veriyi atıyorum. SagaHareket()'te ekleme de yaptığımız gibi geçici düğüm içerisinde ilerleyeceğiz fakat bunu yaparken her ilerlemeden önce geçici düğümün bir ilerisindeki düğümün yığıt top() verisini kendisine atayacağız. Bu sadece bir veriye sahip yığıt için geçerlidir. İçinde birden çok veri olursa aradaki verileri almak için pop() metodu kullanmamız gerekir fakat bu da bizim için veri kaybına sebep olur. Dolayısıyla ikinci bir geçici düğümle daha bu işlemi gerçekleştirdim. SolaHareket() işlemleri SagaHareket() ile aynı sadece bunda işlemimiz geçici düğüm ile ileri gitmek değil geri gitmektir. Yazdırma işlemleri ise aynı geçici bir düğümle listede ilerleyerek yığıt içinde veri kaybetmeden verileri ekrana yazar. Main dosyasında ise DaireselBagilListe class'ından pointer nesne oluşturarak fonksiyonlarımızı menü içinde kullandım. Böylece işlemler tamamlanmış oldu. Bu arada ödevde en çok kafamın karıştığı ve zorlandığım kısım Ekle(veri) komutunu oluşturmaktı.