

Ödevde bağılı liste kullanımı, düğümler ve pointerların daha iyi anlaşılması hedeflenmiştir. Bu doğrultuda yaptığım projede Node, LinkedList olmak üzere iki sınıf kullandım.

Node sınıfındaki değerleri public olarak tutmayı tercih ettim bunun sebebi Node sınıfının herhangi bir kompleks yapıda olmaması ve değerlerin dışarıdan değiştirilmesinin sınıf içinde bir soruna yol açmayacak olmasıdır. Bu sınıf; prev ve next olmak üzere Node sınıfından iki pointer, bir tane veriyi tutacak T sınıfından (T, template'dir) data değişkeni ve integer türünde position değişkenine sahiptir. Node sınıfının constructor'ında data değişkeninin değeri dışarıdan alınır, geriye kalan değerler ise başlangıç değeri için NULL ve 0 atanır. Node sınıfının bir de destructor'ı bulunur. Bu methotta Node nesnesinin heap alanından silindiğine dair bilgi console ekranına yazdırılır.

LinkedList sınıfında Node sınıfından head ve tail olmak üzere iki pointer tanımlanır. Bu pointerların amacı liste üzerinde gezinmemizi sağlamaktır. Listeye ilk eleman eklendiğinde eğer liste boşsa head ve tail bu eklenen değeri gösterir. Daha sonra listeye her eleman eklendiğinde kuyruk değeri son eklenen değeri göstererek listemiz genişletilir. Bu sayede listeye ne kadar eleman eklenirse büyüklüğü de o kadar olur. Bu liste verisinin en güçlü olduğu yönlerinden biridir. LinkedList sınıfı ayrıca size adlı integer türünde bir değişkene sahiptir. Bu değişkene sınıf dışından değer ataması yapılamaz. Başlangıç değeri sıfırdır ve listeye her yeni eleman eklendiğinde bu değişkenin değeri bir arttırılır. Bu değişkenin değerini method yardımı ile listLength() methodu ile döndürülür. Bu methoda sınıf dışından erişim mümkündür. Bir diğer değişken de wordSize'dır. Bu değişken de dışarıdan müdahaleye kapalıdır. Bu değişken, size değişkeninden farklı olarak listeye eklenen farklı kelime sayısını tutar. Sınıfta bir de bool değişkenini döndüren isUnique methodu bulunur. Bu method sınıf dışından çağırılamaz. Böyle tercih etmemin sebebi dışarıdan çağırılmasının bir avantajı bulunmamasıdır. Bu method parametre olarak Node türünde pointer alır. Bu pointer'ın data değerinin daha önce listede bulunup bulunmadığını kontrol eder. Eğer daha önce listede varsa kaç düğüm sonra eklendiğini Node'un içindeki position değerine verir ve false değeri döndürür. Eğer daha önce listede böyle bir kelime kayıt edilmediyse true değeri döndürür. Bir diğer method ise totalEarning'dir. Bu methotta aynı isUnique methodu gibi private olarak tanımlanmıştır. Herhangi bir parametre almaz. Bu method bize listenin daha önce kaydedilen kelimeleri kaydetmeyerek bize ne kadar alan kazandırdığını hesaplar ve döndürür. Bağlı listenin en önemli methodu pushBack'tir. Bu method dışarıdan alınan template olan değişkenin adresini parametre olarak alır. Bu adresi tail değerine ekler. Ekleme yapılırken isUnique methodu sayesinde eşsiz olup olmadığı kontrolü yapılır. Son metodumuz da printAll metodudur. Bu metod düğümlerin yardımıyla listeyi gezerek tek tek listenin değerlerini ekrana bastırır.

Son olarak da program sınıfında yazdığım sınıfları kullanarak programı oluşturdum. Bu program içerik.txt dosyasını fstream paketinin içindeki ifstream nesnesini kullanarak içeriğini okur. While döngüsü ile okunan her kelime listeye tek tek kaydedilir. Liste yazdırıldıktan sonra da heap alanında oluşan nesneler silinir. Ve program sonlandırılır.

Genel olarak bu ödevden öğrendiklerim; C++ Syntax'ı, C++ Sınıf yapısı, Pointers, Bellek Yönetimi ve bağılı liste kavramları oldu.