Sakarya Üniversitesi Bilgisayar Mühendisliği
Veri Yapilari Final Odevi raporu

Istenilenler:

Veri yapilari final odevinde bizden,bilgileri # karakteri ile ayrilmis Kisiler dosyasindan bilgileri okuyarak Kisileri AVL agacina gerekli algoritmayi kullanarak eklememiz,Kisiyi tutan her dugum'e ait stackde,dugumun haraketlerini tutmamiz,agac islemleri bittikden sonra agaci PostOrder olarak okuyarak her dugume ait stackde bulundurduklari verilerle birlikte ekrana yazdirmamiz istenmistir.Program bittiginde copleri temizlememiz istenmistir.

Ogrendiklerim:

Stackde degisen capacity kavramini,stacke ekleme cikarma islemlerini ogrendim.Top metodunun mantigini kavradim.Ama odevin ilerleyen bolumlerinde bu metoda ihtiyacim olmadigina karar vererek daha baska bir yontemle yola devam ettim.Internet uzerindeki avl agaci simulatorlerini kullanmak agaci aklimda canlandirmami ve anlamami daha kolay kildi.Sadece bir sorun vardi arastirdigim kadariyla butun simulatorlarde ayni veriye sahip dugumu ekledigimde yanlis ekliyordu hatta bazi simulatorlerde warning vererek eklemiyordu.Bunun disinda agac veri yapisinda herhangi yerde yeni dugum olusturmayi,tek ve cift cevrimin mantigini ve uygulamasini,agac uzerinde dolasmayi,agaci postorder durumuna getirmeyi,derinlik kavramini ve level kullanarak dugumun onceki ve yeni seviyyesini karsilastirarak stacke gerekli bilgileri atmayi ogrendim.

Odevde yaptiklarim 1.kisim:Okuma

Dosyayi line line okudum,okudugum linelarda # karakterlerini bularak oraya kadar ayirdim ve her satirda bilgi turleri ayni oldugunda ilk #'e kadar olan kismi isme,ikincisini dogum yilina,sonuncusunu kiloya atadim.

Odevde yaptiklarim 2.kisim: Agaca ekleme

Eklerken ilk once Dugum pointeri ve kisi pointeri alan AraVeEkle metodunu cagiriyorum.Bu fonksiyon yeni gelenle altdugume ait yasalari kiyasliyor recursive manitgiyla yaptigim icin herhangi mantiksal sorun olmuyor.Dugum yapisinda constructor kisi pointeri aliyor.Ilk eklenen dugumun basta sagi ve soluna null ataniyor,yukseklik 0 olarak ataniyor ve her dugume ozel yigit kullanmak adina constructorda yigit tanimlamasi yapiyorum.Ilk basta yigita "O" atabilmek icin,yigitin constructorunda gerekli tanimlamari yaptiktan sonra O harfini pusluyorum.Dugum degisikligi lazimsa gerekli degisiklikleri sagcocukiliedegistir veya solcocuk ile degistir fonksiyonlarina yonlediriyorum.Ekleme islemi bittikden sonra derinlik kontrolu yapiyorum.Dugumun derinligini buluyorum bu degeri yeniderilik degiskenine atiyorum.Oncekiderinlik yeni derinlikten buyukse yukselmistir dolayisiyla yigita "Y" harfi

atanacktir eger dugumun onceki derinligi yenisinden kucukse alcalmistir yigita “A” harfi atanacktir.Bu iki kontrole de girmediyse demektir ki dugumde degisiklik olmamis yigita “D” harfi atanacaktir.Yigita atamalari Push metodunu kullanarak yapilmaktadir.Kontrol fonksiyonum recurusivedir dugum eklendikten sonra dugumun sagindaki sonlundaki derinligi her eklemede kontrol edecektir.Alcalma yukselme islemleri bittikden sonra DerinlikEsitle fonskiyonum yaridimiyla dugumun derinligini guncelliyorum ki bir sonraki eklemede dugumun onceki derinligiyle hesap yapilmasin.DerinlikEsitle fonksiyonumun algoritmasi recurusive oldugu icin dugumun derinliginde bir degisiklik oldugu zaman alt dugumlerini de etkilmesinden dolayi dugumun sag ve solundaki dugumlerin derinliklerini de guncelliyorum.

Odevde yaptiklarim 3.kisim: PostOrder olarak okuma ve ekrana yazdirma

Agac islemleri bittikden sonra,agaci postorder olarak,yeni ilk once solcocugu sonra sag cocugu en sonda root’u okuyarak yazdirdim.Stackdeki verileri yazdirirken Stackin kendi “Pop()” metodunu kullandim.

Eksik biraktigim yerler:

Gozlemledigim ve test ettigim kadariyla odevimde eksik bir kisim yok.

Odevde zorlandigim kisimlar:

Dosyayi satir satir okuyup bilgileri ayirmakta zorlandim.Satirlari okumakta problem yoktu ama onlari # yardimiyla ayirmakta ve satir her dondugunde bu bilgileri Kisinin gerekli alanlarina atmakta problem yasiyordum.Sonra aslinda cok basit oldugunu farkettim yapmam gereken tek sey getline’la ayridiktan sonra delimiter tanimlayip bilgileri ayirmakti.Bir tane delimiter daha tanimlayarak satirin nerede bittigini ogrene bilecektim.Onun disinda kisiyi agaca ekledikten sonra,seviyesini bulurken zorlandim onu da internetten aldigim kodu modifiye ederek cozdum.Bu kod benim derinlikbulma algoritmamin ana temelini olusturuyordu.Buradan yola cikarak eger derinligi buluyorsam,onceki derinligi ile karsilatirmam gerekiyor dugum haraketlerini gorebilmek icin en sonda da degisim olduysa derinligini guncellemem gerekiyor.Bunlarin algoritmalarini yazdiktan sonra da odev bitmis oldu.

Hazirlayan:Maftun Hashimli. G181210554