

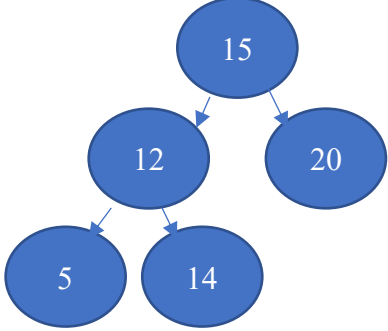
BLM2512 Veri Yapıları ve Algoritmalar Ödev – 2

Konu : Binary Search Tree Üzerinde Meyve Toplama Oyunu

Problem: 1...N arasındaki sayılar kullanılarak rastgele olarak üretilen **BST özelliği taşıyan** M adet düğümden oluşan bir ağaçta yer alan meyveleri düşürmek için atılan **taşlar k değerine sahip meyveye** isabet ettiğinde **atışın gücüne (p)** bağlı olarak meyvenin ağaca tutunmasını zayıflatmaktadır. Her atış sonunda isabet alan meyvenin tutunma gücü azalmakta fakat ağaçtaki yeri değişmemektedir. k değerine sahip meyvenin **tutunma gücü 0 ve altına düşerse** ağaçtan yerdeki kovaya düşmektedir. Herhangi bir **meyve düştüğünde ağacın BST özelliği bozulduğu** için ağaç yeniden düzenlenmelidir. Atışlar sonunda ağaçta meyve kalmadığında kovaya düşen M adet meyve **kovaya düşme sırasına** göre ekrana yazdırılmalıdır. Kovanın “**Stack**” yapısında olduğu varsayılmalıdır. İhtiyaç duyulması halinde ikinci bir boş kova kullanılabilir.

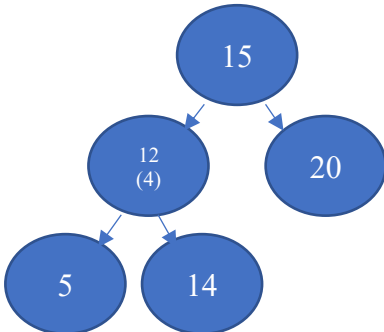
Örnek:

N=20, M=5 için,

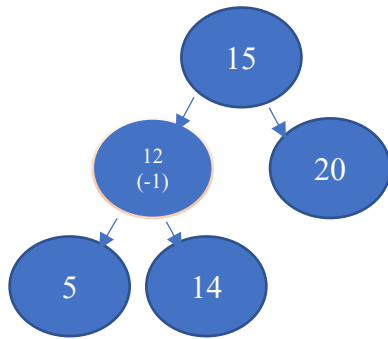


Atış:

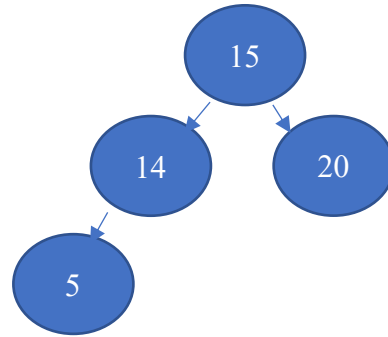
k:12, p=8



Atış:
k:12, p=5

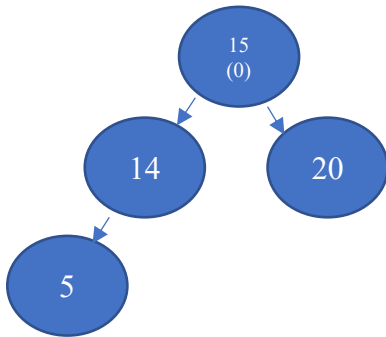


->

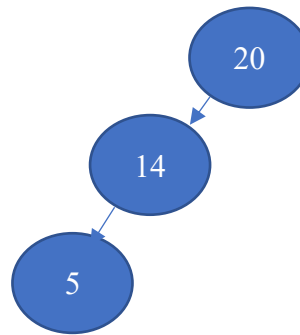


Kova: 12

Atış:
k:15, p=15



->



Kova: 15 12

.....

Kova: 14 20 5 15 12

Çıktı: 12, 15, 5, 20, 14

Ödevde Gerçekleştirilecekler:

- 1) Giriş bilgisi olarak kullanıcıdan N (maksimum meyve ağırlığı) ve M (meyve adedi) değerlerini alınız.
- 2) BST için structure yapısını tanımlayınız.
- 3) BST'yi rastgele üreteceğiniz değerler ile oluşturunuz. Ağacı üretirken linkli liste veri yapısından faydalanınız.
- 4) Her turda ağacın son halini gösteriniz.
- 5) Her turda kullanıcıdan vurmak istediği meyvenin değerini (k) ve atış gücünü (p) alınız. Atışlar her turda farklı meyvelere isabet edebilir. Düşene kadar hep aynı meyve vurulmak zorunda değildir.
- 6) BST yapısının bozulmaması için meyve düşene kadar ilk değerini kaybetmeyiniz.
- 7) Eğer meyve daldan kovaya düşerse BST ağacını yeniden kurallara uygun olarak organize ediniz.
- 8) Tüm meyveler kovaya düştüğünde meyvelerin ilk değerlerini (k) kovaya düşme sırası ile ekrana yazdırınız. Kova işlemlerinde sadece stack fonksiyonları kullanılabilir.

Ödev Teslimi ile ilgili önemli bilgiler:

Aşağıda verilen bütün bilgileri içeren tek bir doküman hazırlayarak **18.04.2023 saat 23:59'a** kadar online.yildiz.edu.tr adresi üzerinde tanımlı ödev **OgrenciNumarasi.rar** veya **OgrenciNumarasi.zip** dosyasını yükleyiniz.

Geç Ödev Yükleme Koşulları :

Herhangi bir nedenle gecikme yaşayan öğrencilerin mağdur olmamaları için 19.04.2023 saat 00:01 ve 00:30 arası geç ödev yükleme alanı aktif edilecektir. Bu alandan yükleme yapan öğrencilerin ödevleri 20 puan kesinti ile 80 üzerinden değerlendirilecektir. 30 dakikadan fazla gecikme yaşayan öğrencilerin ödevleri değerlendirilmeyecektir.

Teslim Edilecekler:

1. Anlatılan problemi çözen linkli liste veri yapısını ve ilgili fonksiyonları içeren programı **C** dilinde yazarak **OgrenciNumarasi.c** dosyasını yükleyiniz.
2. Farklı senaryoların bulunduğu ekran çıktılarını içeren dosyayı **OgrenciNumarasi.pdf** formatında yükleyiniz.
3. Kısa bir **video** (5-10 dk.) hazırlayınız. Video içeriğinde problemi, geliştirdiğiniz çözümü ve kodunuzu anlatınız. Ardından farklı girdiler ile programınızın çalışmasını gösteriniz. Video linkini raporunuza ekleyiniz. Video linkini public paylaşmayınız, kopyaya sebep olmaktadır.

Teslim Edilecek Dokümanlar:

- HW#_OgrenciNumarasi.zip (Örn: HW3_15011001.zip)
 - OgrenciNumarasi.pdf (Örn: 15011001.pdf)

- ÖğrenciNumarasi.c (Örn: 15011001.c)
- Uygulama video linki

- VERİLEN SÜRE DOSYA YÜKLEME İŞLEMLERİNİ DE KAPSAMAKTADIR.
- E-POSTA ile GÖNDERİLEN CEVAPLAR KESİNLİKLE DEĞERLENDİRİLMEYECEKTİR.
- DOSYA FORMATINA ve İSİMLENDİRME KURALLARINA UYULMADIĞI TAKDİRDE SORU PUANIN YARISI ÜZERİNDEN DEĞERLENDİRECEKTİR.
- DOSYA DIŞINDA SİSTEME HERHANGİ BİR **DRIVE LİNK**'i EKLEMİYİNİZ.
- BAŞKA BİR ÖDEVE VEYA İNTERNET ÜZERİNDE BULUNAN BİR ÇÖZÜME BENZERLİĞİ YÜKSEK OLAN ÖDEVLER KOPYA OLARAK DEĞERLENDİRİLECEKTİR.