



Ad & OgrNo: Hayyam Islamzade G141210098

Ad & OgrNo: Ayşenur Odabaşı G161210094

Ödev

Ödev ilk açıldığında kullanıcıdan bir sayı istenmektedir. Girilen sayının basamak uzunluğu kadar basamak yüksekliğine bağlı olarak arama ağacında düğümler oluşturulacaktır. Arama ağacından en küçük değer silinmek istendiğinde tüm arama ağaçları içerisinde sadece sol tarafda olan düğümler(Arama ağacının çalışma prensibi , küçük değerler solda büyük değerler sağda olacak şekildedir)arasında en küçük değeri bulacak ve o değeri sildikten sonra arama ağacı yapısını düzeltecektir. Arama ağacından en büyük değer silinmek istendiğinde tüm arama ağaçları içerisinde sadece sağ tarafda olan düğümler arasında en büyük değeri bulacak ve o değeri sildikten sonra arama ağacı yapısını düzeltecektir.

Girilen değer yüksekliğine bağlı olarak ağac içerisinde rastgele sayı oluşturmak

```
for (int i = 0; i < number.length(); i++)
{
    int x = (int)number[i] - 48;
    list->Add(x);
}
```

Yukardaki kodda girilen number değişkeninde bulunan basamak için list içerisindeki Add fonksiyonuna yükseklik parametresini gönderir.

```
while ((listMover.current->subTree->Height() < height))
{
    random_number = (rand() % 999) + 1;
    listMover.current->subTree->Add(random_number);
}
```

Daha sonra ağaç yüksekliği gelen yükseklik değerine eşit olana kadar ağac içerisine random sayı ekler

Min Değer Silme

```
for (; !listMover.isLast(); listMover.next()){
    if (min > listMover.current->FindMinNode())
    {
        min = listMover.current->FindMinNode();
        tree = listMover.current;
    }
}
cout << "silinen min : " << min;
if (tree->Height() > 1)
    tree->Delete(min);
```

```

else
{
    tree->Delete(min);

    Delete(tree);
}

```

Yukarıki kodda tüm ağaçlar içerisinde gezinerek min değeri ve bu min değerinin bulunduğu ağacı bulur. Daha sonra eğer ağaç yüksekliği 1 den büyükse (kök değeri varsa) sadece değeri siler eğer değilse tüm ağacı siler.

Max Değer Silme

```

for (; !listMover.isLast(); listMover.next()){
    if (min < listMover.current->FindMinNode())
    {
        min = listMover.current->FindMinNode();
        tree = listMover.current;
    }
}
cout << "silinen min : " << min;
if (tree->Height() > 1)
    tree->Delete(min);
else
{
    tree->Delete(min);

    Delete(tree);
}

```

Yukarıki koddada min silme işleminde olduğu gibi bir yapı vardır. Tek farkı büyük değeri siler .