

YILDIZ TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



2018-2019 BAHAR YARIYILI
BLM2512 VERİ YAPILARI VE ALGORİTMALAR DERSİ
GRUP-2
3.ÖDEV RAPORU

Hazırlayan: Mehmet Hayri Çakır – 16011023

Dersin Yürütücüsü: Dr.Öğr.Üyesi Göksel Biricik

İstanbul, Mayıs 2019

İçindekiler

1. Yöntem.....	2
1.1. Problem.....	2
1.2. Akış diyagramları.....	2
2. Uygulama	4
2.1. Rastgele Sayılarla Örnek.....	4
2.2. Küçükten Büyüğe Sıralı Sayılarla Örnek	5
2.3. Büyükten Küçüğe Sıralı Sayılarla Örnek	6
3. Sonuç.....	7

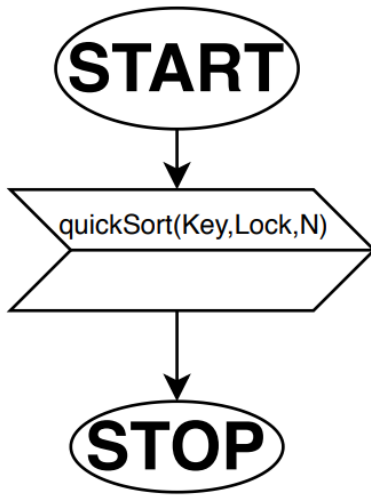
1. Yöntem

1.1. Problem

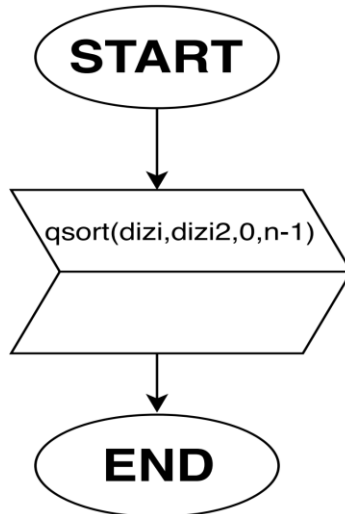
Bir kulede bulunan N odayı açmak için farklı büyüklükte N adet anahtar gerekiyor. Anahtarları diğer anahtarlarla veya kilitlerin büyüklüklerini diğer kilitlerle karşılaştırmadan, sadece anahtarlar kapılara takılarak kontrol yapılabiliyor. Her seferinde rasgele bir anahtar seçip hangi kilide uyduğunu bularak, anahtarların uydukları bütün kapıları bulan algoritmayı tasarlamak.

1.2. Akış diyagramları

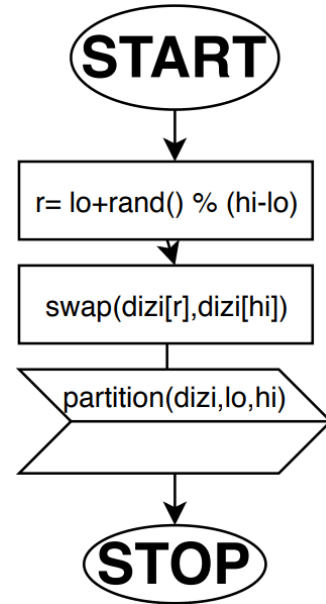
main fonksiyon



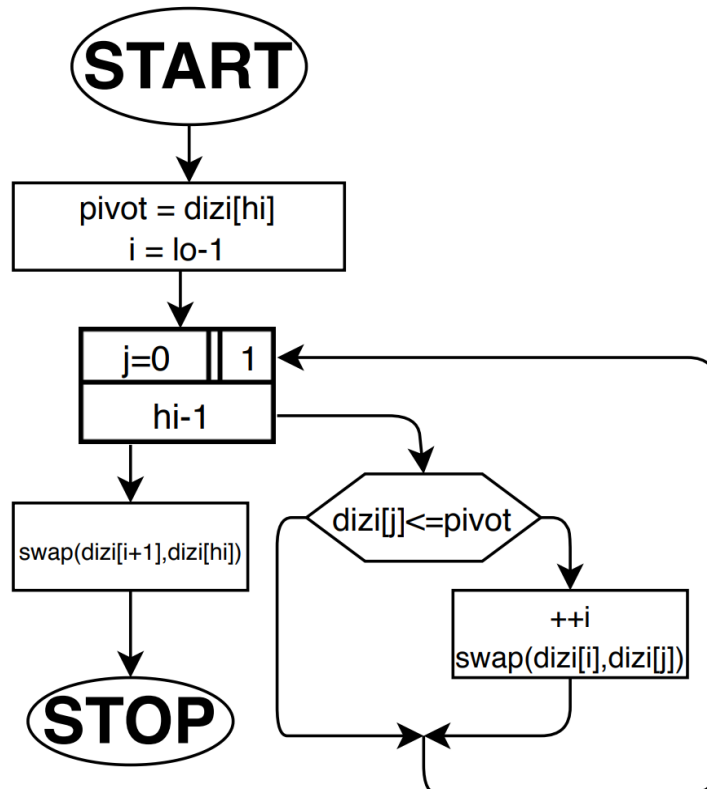
quickSort fonksiyonu



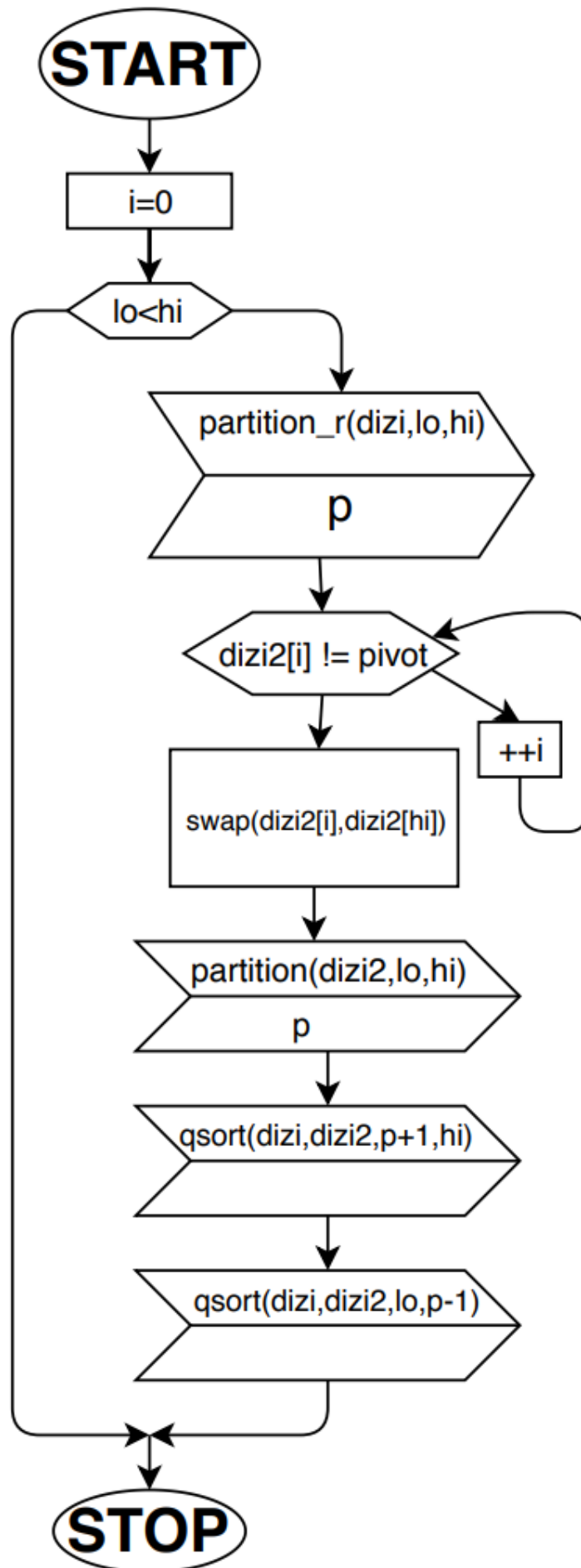
partition_r fonksiyonu



partition fonksiyonu



qsort fonksiyonu



2. Uygulama

2.1. Rastgele Sayılarla Örnek

```
=====
Key[N]:      2 4 1 3 6 5
Lock[N]:     6 3 2 5 1 4
=====

Pivot: 1
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 4 5 3 6 2
Lock[N]:     6 3 2 5 1 4
-----

Pivot: 1
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 4 5 3 6 2
Lock[N]:     1 3 2 5 4 6
=====

Pivot: 4
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 6 5
Lock[N]:     1 3 2 5 4 6
-----

Pivot: 4
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 6 5
Lock[N]:     1 3 2 4 6 5
=====

Pivot: 6
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 3 2 4 6 5
-----

Pivot: 6
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 3 2 4 5 6
=====

Pivot: 2
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 3 2 4 5 6
-----

Pivot: 2
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
=====

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
=====
Press any key to continue . . .
```

2.2. Küçükten Büyüğe Sıralı Sayılarla Örnek

```
=====
Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
=====
Pivot: 1
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
-----
Pivot: 1
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
-----
Pivot: 3
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
-----
Pivot: 3
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
-----
Pivot: 5
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
-----
Pivot: 5
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
-----
Press any key to continue . . .
```

2.3. Büyükten Küçüğe Sıralı Sayılarla Örnek

```
=====
Key[N]:      6 5 4 3 2 1
Lock[N]:     6 5 4 3 2 1
=====

Pivot: 5
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 4 3 2 5 6
Lock[N]:     6 5 4 3 2 1
-----

Pivot: 5
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 4 3 2 5 6
Lock[N]:     1 4 3 2 5 6
=====

Pivot: 1
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 4 3 2 5 6
Lock[N]:     1 4 3 2 5 6
-----

Pivot: 1
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 4 3 2 5 6
Lock[N]:     1 4 3 2 5 6
=====

Pivot: 3
Yazdirilan pivota gore key dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 4 3 2 5 6
-----

Pivot: 3
Yazdirilan pivota gore lock dizisi duzenleniyor...

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
=====

Key[N]:      1 2 3 4 5 6
Lock[N]:     1 2 3 4 5 6
=====
Press any key to continue . . .
```

3. Sonuç

Kötü durumda, her partition işlemi dizinin bir tarafında $N/4$, diğer tarafında $3N/4$ eleman kalacak şekilde gerçekleşir. Kötü durum rekürsif ağacı $O(\log N)$ olur.

$$T(n) < T(n/4) + T(3n/4) + O(n)$$

$$T(n) < 2T(3n/4) + O(n)$$

$$\text{Sonuç} = O(N \cdot \log N)$$