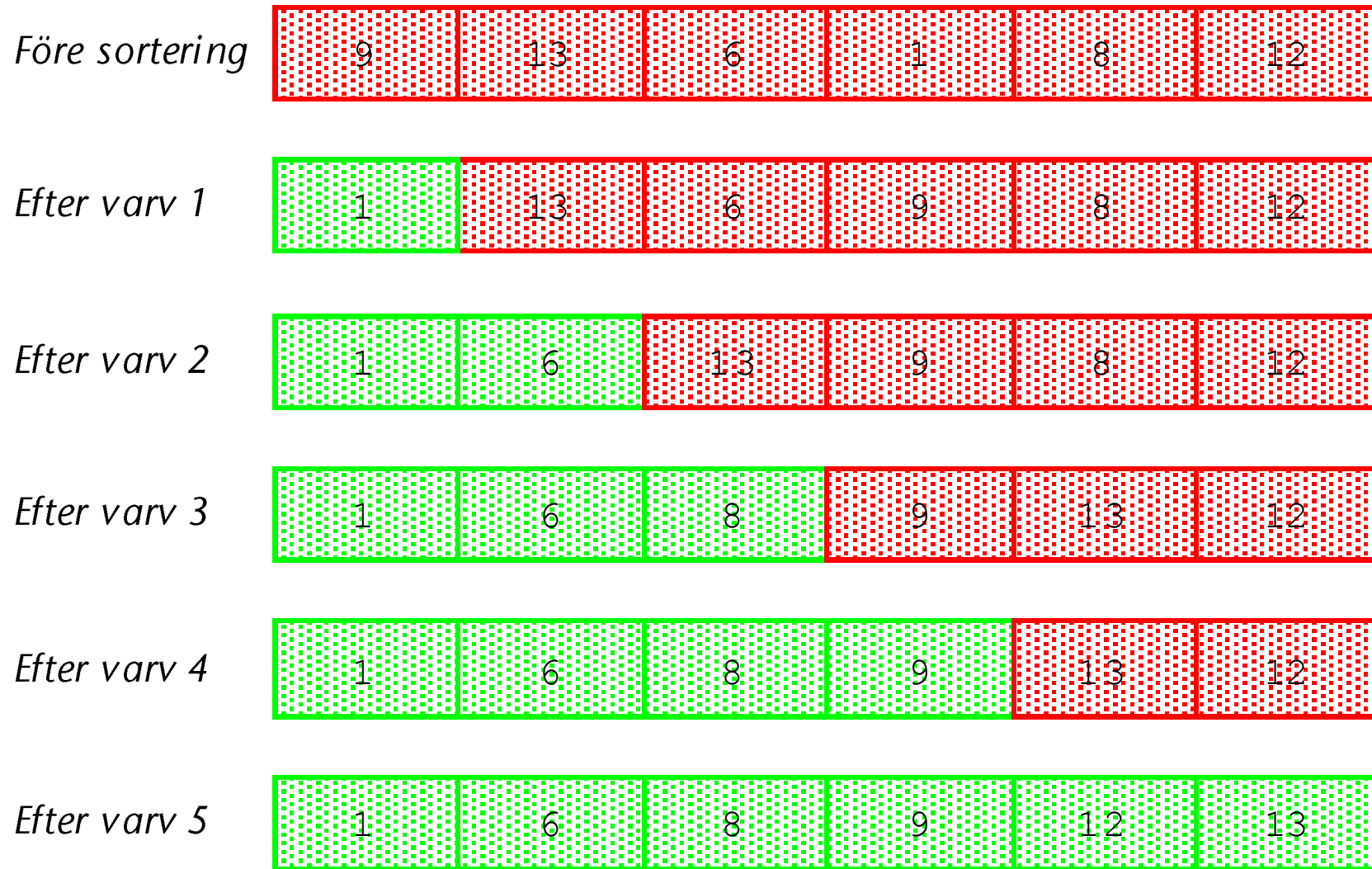


## Sortering, selection sort ("sök minsta och byt")



```
public static void sortera(double[] a, int start, int slut) {  
    for (int k=start; k < slut; k++) {  
        // Sök den minsta bland komponenterna nr k till slut  
        int m = k; // index för hittills minsta värde  
        for (int i=k; i<=slut; i++)  
            if (a[i] < a[m])  
                m = i; // nytt minsta värde funnet  
        // Låt komponenterna nr k och m byta plats  
        double temp = a[k];  
        a[k] = a[m];  
        a[m] = temp;  
    }  
}
```

```

public class Person {
    String förnamn, efternamn;
    int föddÅr;
    boolean singel;
}

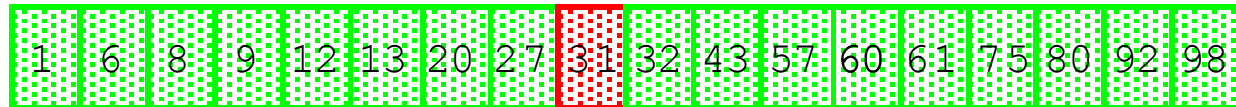
public static void sortera(Person[] a, int start, int slut){
    Collator c = Collator.getInstance();
    c.setStrength(Collator.PRIMARY);
    for (int k=start; k < slut; k++) {
        // Sök den minsta bland komponenterna nr k till slut
        int m = k; // index för hittills minsta värde
        for (int i=k; i<=slut; i++)
            if (c.compare(a[i].efternamn, a[m].efternamn) < 0 ||
                (c.compare(a[i].efternamn, a[m].efternamn) == 0 &&
                 c.compare(a[i].förnamn, a[m].förnamn) < 0))
                m = i; // nytt minsta värde funnet
        // Låt komponenterna nr k och m byta plats
        Person temp = a[k];
        a[k] = a[m]; a[m] = temp;
    }
}

```

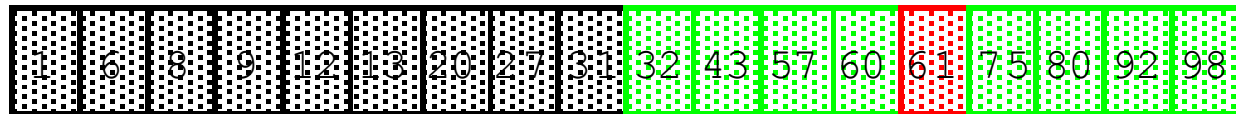
## Binärsökning

Exempel: Sök efter talet 60

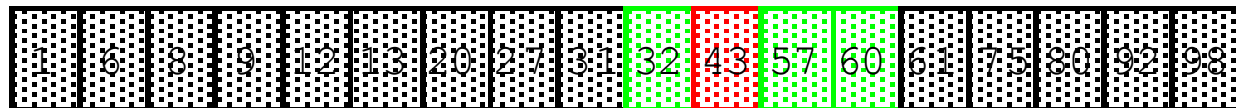
Rad 1



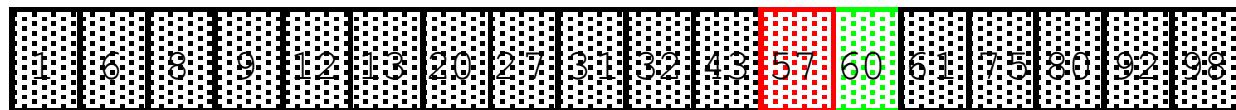
Rad 2



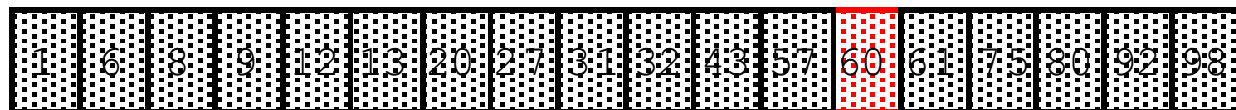
Rad 3



Rad 4



Rad 5



```
public static int binSök(double[] a, int start, int slut, double sökt) {  
    int i1 = start, i2 = slut, m = 0;  
    while (i1 <= i2) {  
        m = (i1 + i2)/2;    // beräkna index för mitten  
        if (sökt < a[m])  
            i2 = m-1;  
        else if(sökt > a[m])  
            i1 = m+1;  
        else // lika  
            break; // man hittade talet  
    }  
    if (sökt == a[m])  
        return m;  
    else  
        return start-1;  
}
```