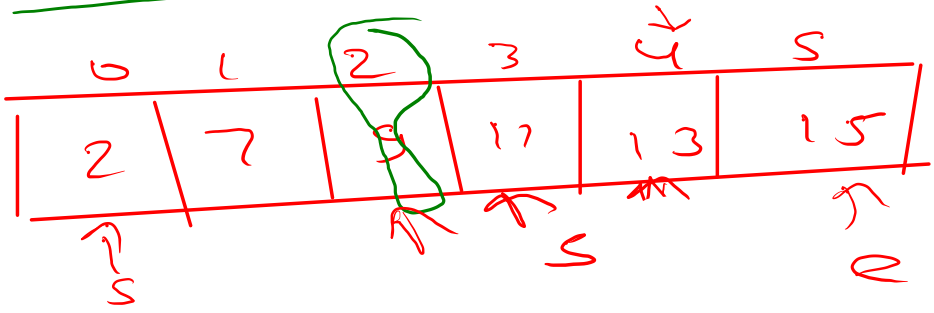
Binary search

tar = 13



$\frac{9 == 13}{9 < }$

(1)

$\frac{0+5}{2} = 2.5$

$\frac{s+(e-s)}{2}$

$\frac{3+5}{2} = \frac{8}{2} = 4$

$\frac{2}{2}$

$\Rightarrow \quad 2s + (e-s) \Rightarrow \frac{2s+e-s}{2}$

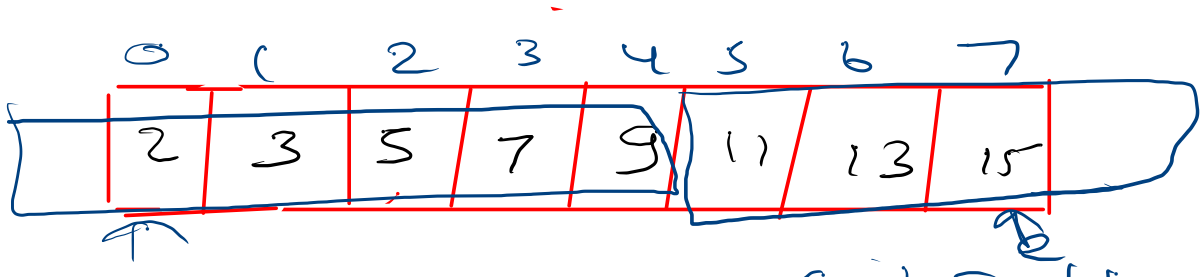
$= \frac{s+e}{2}$

Square root

16

tar = 10

ceil
floor



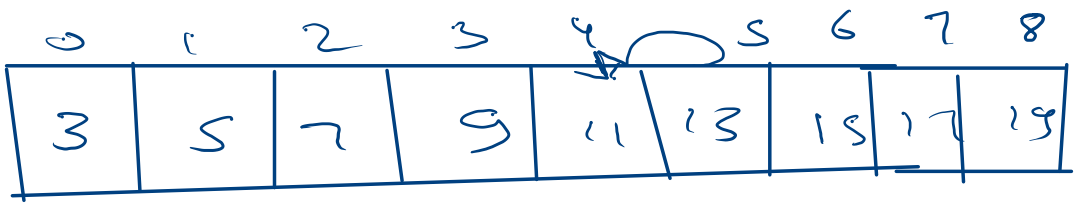
s

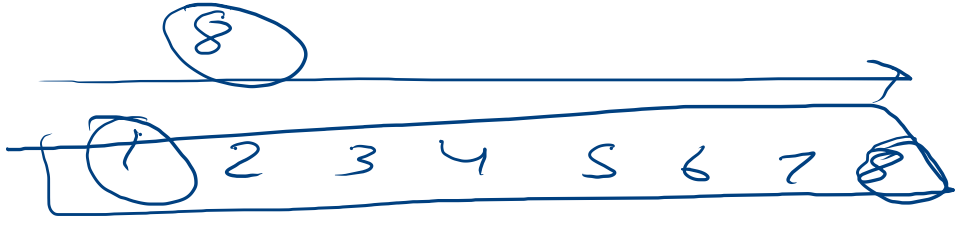
ceil = 11

floor = 9

8 → 2.828 → 2

key = 13





S → 1

E → 7

mid = (s+e) / 2

$\frac{0+8}{2} = 4$

mid

4 × 4 → 16

while (start <= end)

int mid = (s + e) / 2

0 ---- 8

```java
public static int squareRoot(int n){
    int s=1, end=n,ans=-1;
    while(s<=end){
        int mid = (s+end)/2;
        if(mid*mid==n){
            ans=mid;
        }else if(mid*mid<n){
            s = mid+1;
            ans = mid;
        }else{
            end = mid-1;
        }
    }
    return ans;
}
```

n = 8

floor

ceil

1 <= 8

4 x 4 == 8

16 == 8

2 x 2 == 8

4 == 8

s = mid + 1

ans = 3 + 3 / 2

6 / 2 = 3

n = 8

1 2 3 4 5 6 7 8

s

e

ceil

floor

-2^{31} to 2^{31}

int

$a \times b = c$

mid x mid

$2^{31} \times 2^{31}$

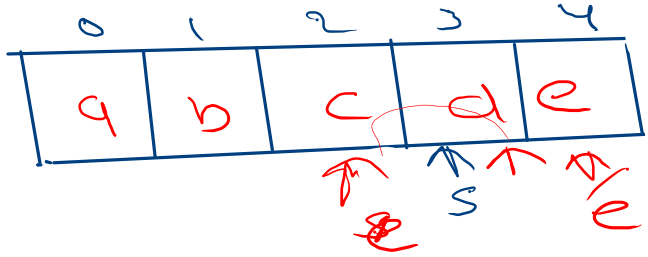
mid

$a = \frac{b}{c}$

$a \times b = c$

$a = \frac{c}{b}$

```
c
5
a b c d e
```



Find Square Root

Language: Java 7 Open in editor

```java
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        System.out.print(squareRoot(n));

    }
    public static int squareRoot(int n){
        int s=1, end=n,ans=-1;
        while(s<=end){
            int mid = (s+end)/2;
            if(mid*mid==n){
                ans=mid;
                break;
            }else if(mid*mid<n){
                s = mid+1;
                ans = mid;
            }else{
                end = mid-1;
            }
        }
        return ans;
    }
}
```

Search Character

Language: Java 7

Open in editor

```java
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
         Scanner sc = new Scanner(System.in);
         char tar = sc.next().charAt(0);
        int n = sc.nextInt();
        char[] ch = new char[n];
        for(int i=0;i<n;i++){
            ch[i] = sc.next().charAt(0);
        }
     searchCharacter(ch,tar,n);
    }
    public static void searchCharacter(char[] ch,char tar,int n){
        int s=0, end = n-1;
        char ans = '$';
        while(s<=end){
            int mid = (s+end)/2;
            if(ch[mid]<=tar){
                s=mid+1;
            }
            else{
                end = mid-1;
                ans=ch[mid];
            }
        }
        if(ans=='$'){
           System.out.println(-1);
        }else{
             System.out.println(ans);
        }

    }
}
```

744. Find Smallest Letter Greater Than Target

```java
class Solution {
    public char nextGreatestLetter(char[] ch, char tar) {
        int n = ch.length;
        int s=0, end = n-1;
        char ans = '$';
        while(s<=end){
            int mid = (s+end)/2;
            if(ch[mid]<=tar){
                s=mid+1;
            }
            else{
                end = mid-1;
                ans=ch[mid];
            }

        }
        if(ans=='$'){
            return ch[0];
        }
        return ans;
}
}
```