

Binary search

$$\log = 13$$

0	1	2	3	4	5
2	7	9	11	13	15

\uparrow s \uparrow e
 \uparrow s \uparrow e

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

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

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

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

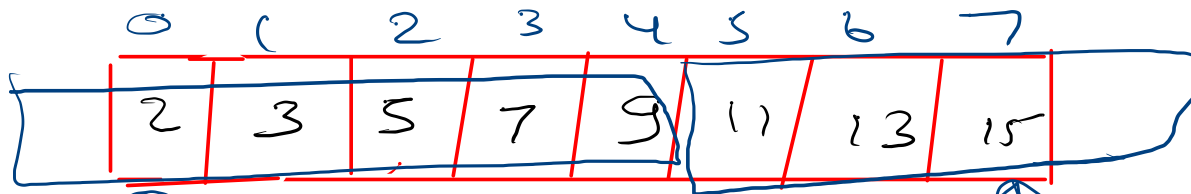
$$2s + (e - s) \Rightarrow \frac{2s + e - s}{2} = \frac{s + e}{2}$$

Half square root

16

$$\text{target} = 10$$

$$\frac{\text{ceil}}{\text{floor}}$$



↑
S

↑
ceil = 11

floor = 9

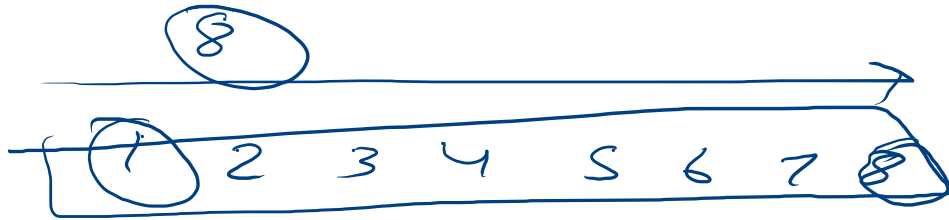
8 →

2.888

→ 2

$$-198 = 132$$

0	1	2	3	4	5	6	7	8
3	5	7	9	11	13	15	17	19



$$S \rightarrow 1$$

$$E \rightarrow$$

$$mid = (S + E)$$

$$\frac{2}{2}$$

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

mid

$$4 \times 4 \rightarrow 16$$

while (start <= end)

0 --- 8

int mid = $\frac{(start + end)}{2}$

```

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;
}

```

$$n = 8$$

~~2.5~~ ceil
61008

$$1 < 8$$

$$4 \times 4 = 8$$

$$16 = 8$$

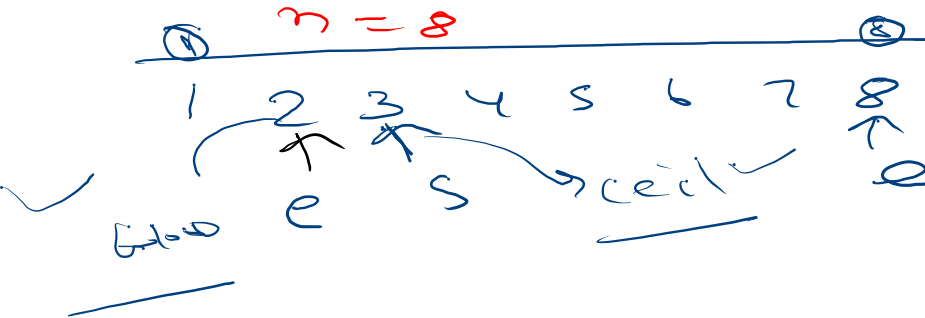
$$2 \times 2 = 8$$

$$4 = 8$$

$$s = \text{mid} + 1$$

$$\text{ans} = \frac{3 + 3}{2}$$

$$\frac{6}{2} = 3$$



⑦

2^{31} to 2^{31}

total

mid x mid

$$a \times b = c$$

mid

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

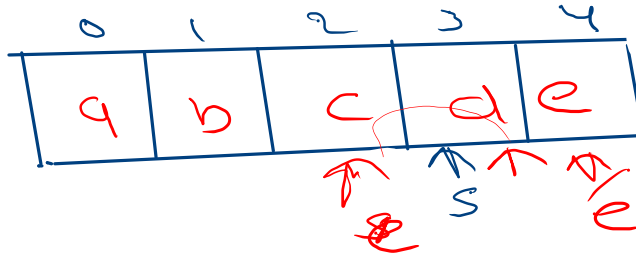
$$a = \frac{b}{c}$$

$$a \times b = c$$

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

c
5
a b c d e

18
8-12



$$3 + 4 = \frac{7}{2} \quad (3)$$

Start
ans = mid
end = mid - 1
mid = 2

7

mid = 4
start =

Find Square Root

Language: Java 7

 Open in editor

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11         Scanner sc = new Scanner(System.in);
12         int n = sc.nextInt();
13         System.out.print(squareRoot(n));
14     }
15
16     public static int squareRoot(int n){
17         int s=1, end=n,ans=-1;
18         while(s<=end){
19             int mid = (s+end)/2;
20             if(mid*mid==n){
21                 ans=mid;
22                 break;
23             }else if(mid*mid<n){
24                 s = mid+1;
25                 ans = mid;
26             }else{
27                 end = mid-1;
28             }
29         }
30         return ans;
31     }
32 }
```


Search Character

Language: Java 7

[Open in editor](#)

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11         Scanner sc = new Scanner(System.in);
12         char tar = sc.next().charAt(0);
13         int n = sc.nextInt();
14         char[] ch = new char[n];
15         for(int i=0;i<n;i++){
16             ch[i] = sc.next().charAt(0);
17         }
18         searchCharacter(ch,tar,n);
19     }
20     public static void searchCharacter(char[] ch,char tar,int n){
21         int s=0, end = n-1;
22         char ans = '$';
23         while(s<=end){
24             int mid = (s+end)/2;
25             if(ch[mid]<=tar){
26                 s=mid+1;
27             }
28             else{
29                 end = mid-1;
30                 ans=ch[mid];
31             }
32         }
33         if(ans=='$'){
34             System.out.println(-1);
35         }else{
36             System.out.println(ans);
37         }
38     }
39 }
40 }
```

744. Find Smallest Letter Greater Than Target

```
1 class Solution {
2     public char nextGreatestLetter(char[] ch, char tar) {
3         int n = ch.length;
4         int s=0, end = n-1;
5         char ans = '$';
6         while(s<=end){
7             int mid = (s+end)/2;
8             if(ch[mid]<=tar){
9                 s=mid+1;
10            }
11            else{
12                end = mid-1;
13                ans=ch[mid];
14            }
15        }
16        if(ans=='$'){
17            return ch[0];
18        }
19        return ans;
20    }
21 }
```