Data Structures

Searching Technique – Binary Search

CODE FORTHINGS

CODE FOR THINGS

Algorithm - Binary Search Iterative

Algorithm

Binary_search_iterative(arr, key)

Input Specification:

arr : Array to hold the elements
key : Item to be searched

Output Specification:

mid : Positon of key element

-1 : Key is not found

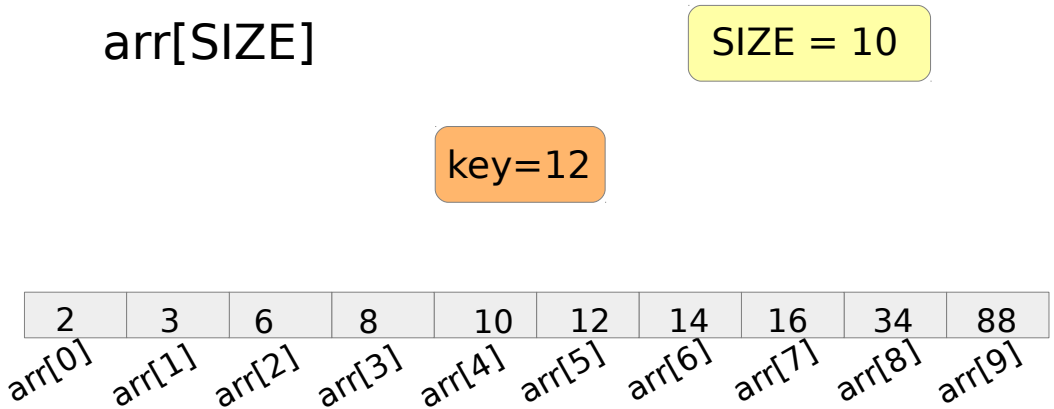
Data Structure – Algorithm

Binary_search_iterative(arr,key)

arr[SIZE]

SIZE = 10

Binary_search_iterative(arr,key)



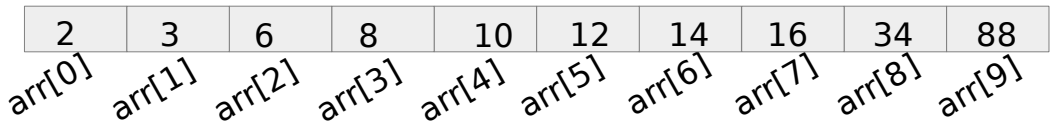
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

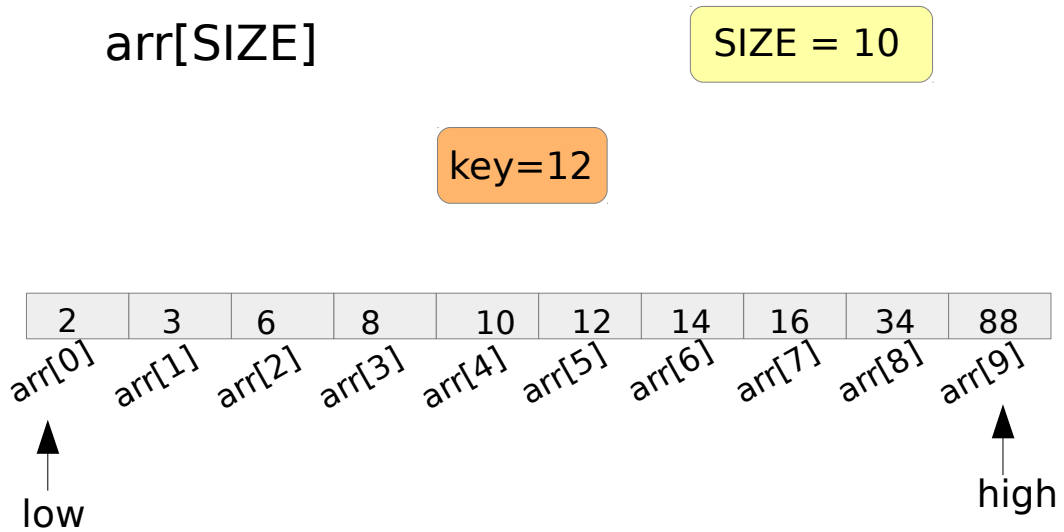
SIZE = 10

key=12



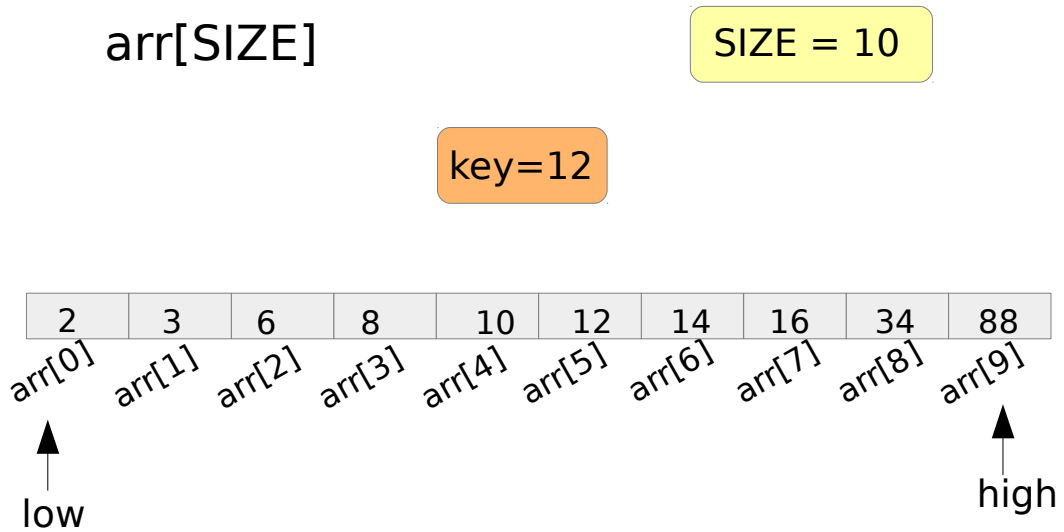
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



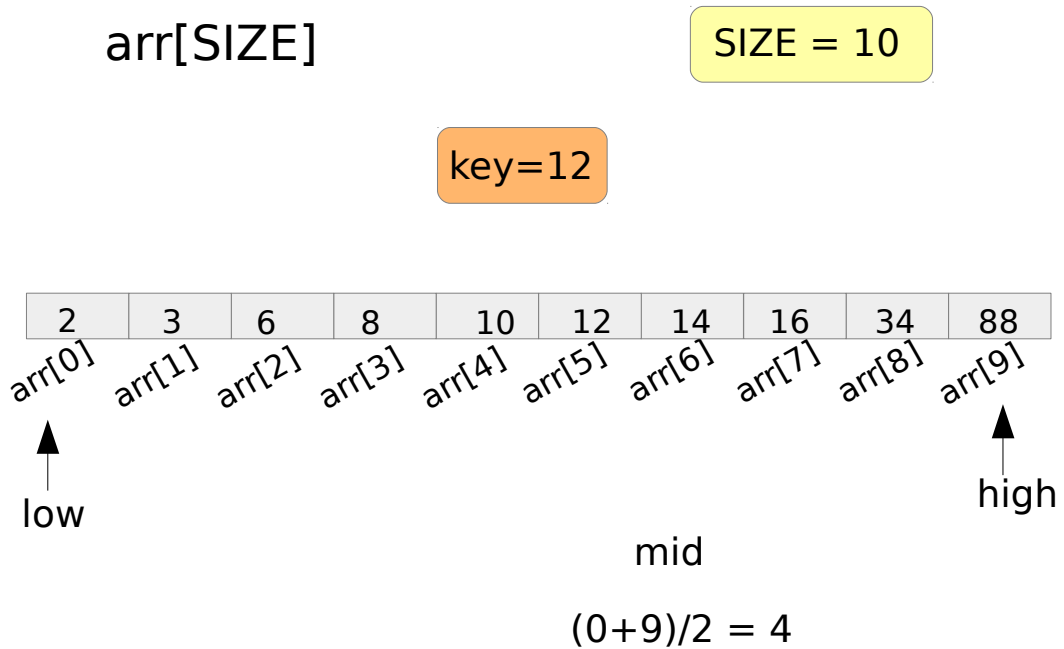
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



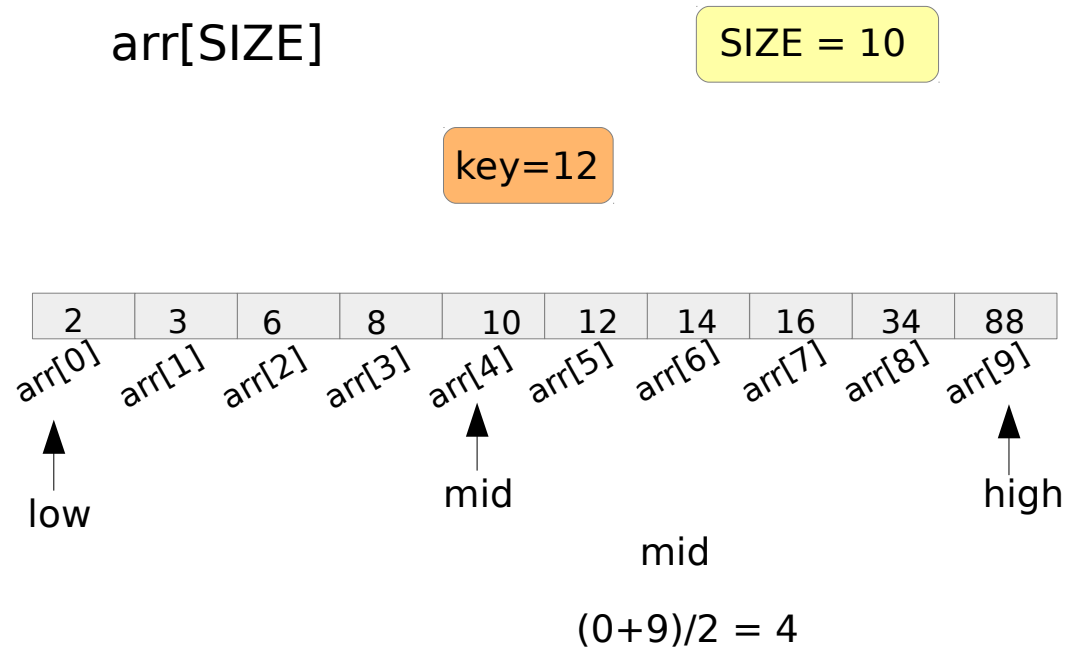
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



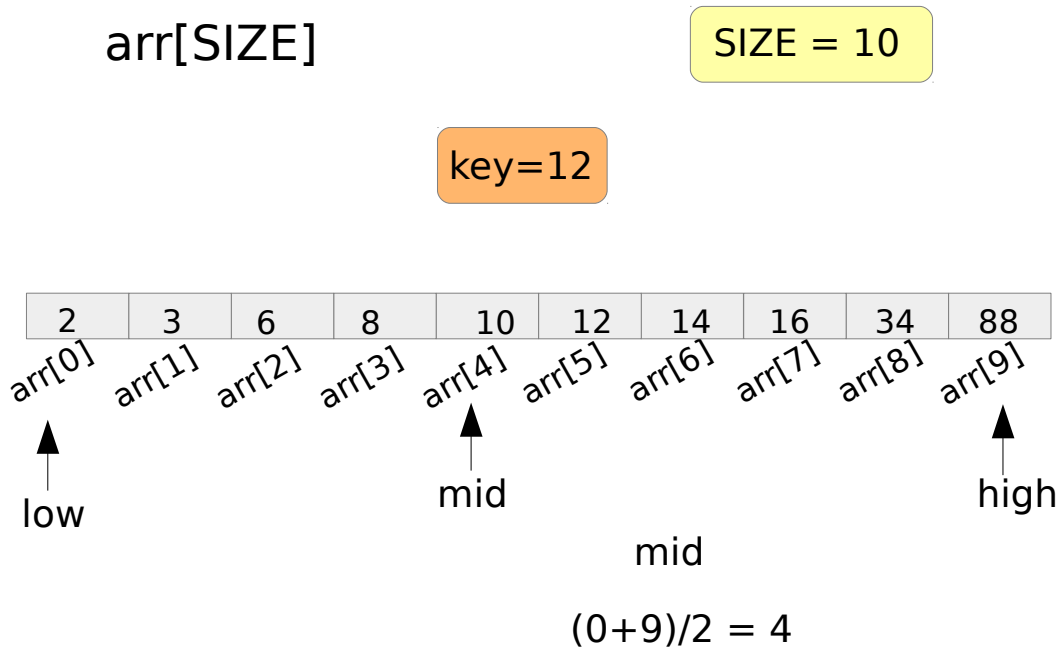
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```



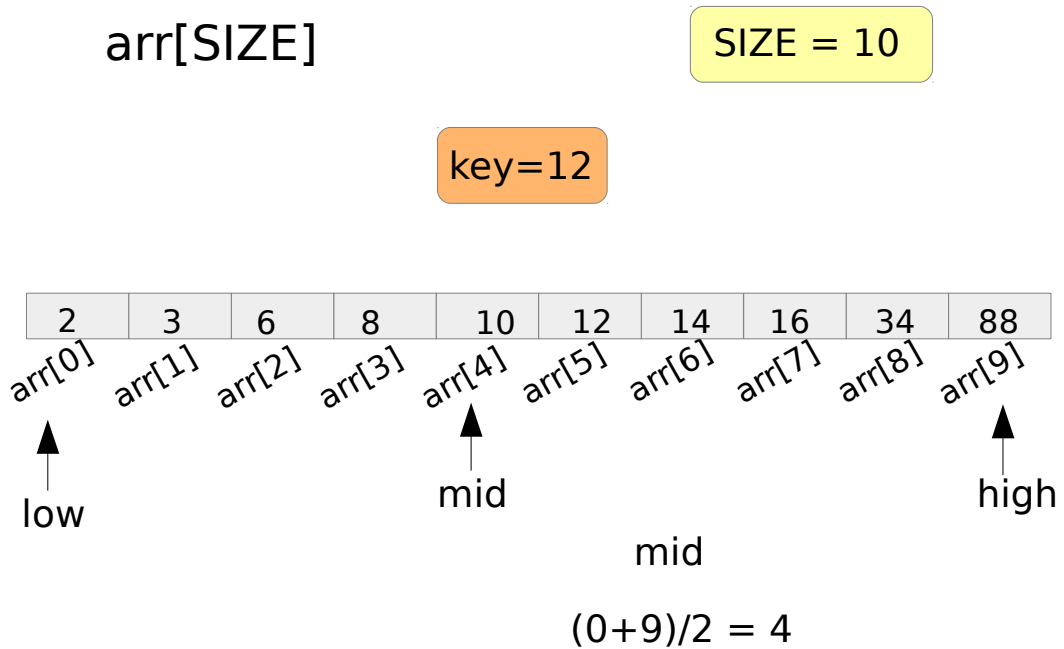
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



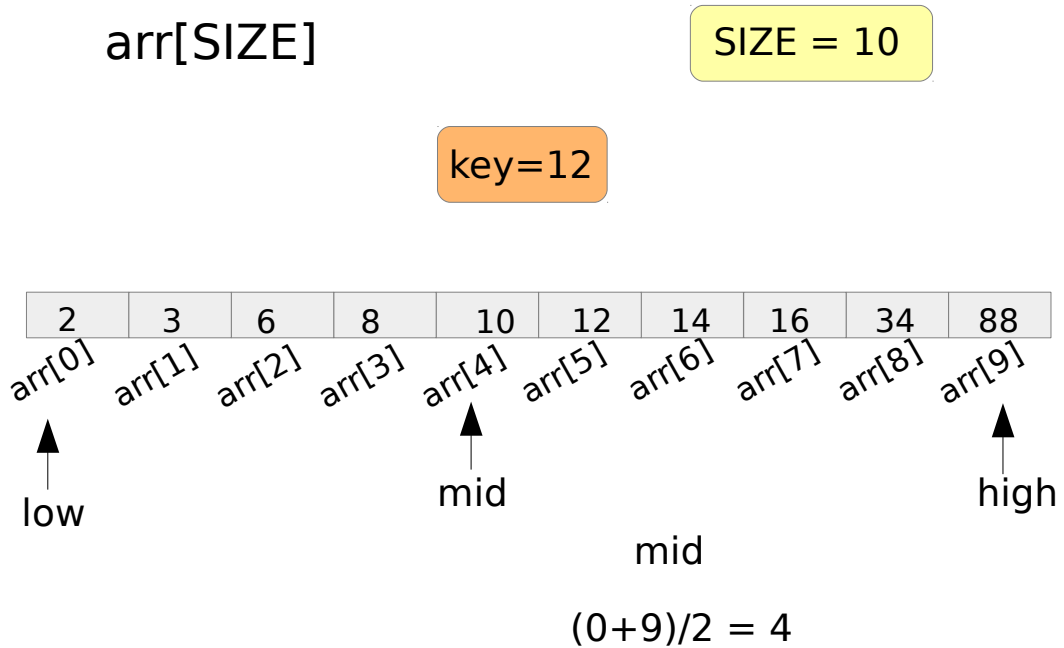
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



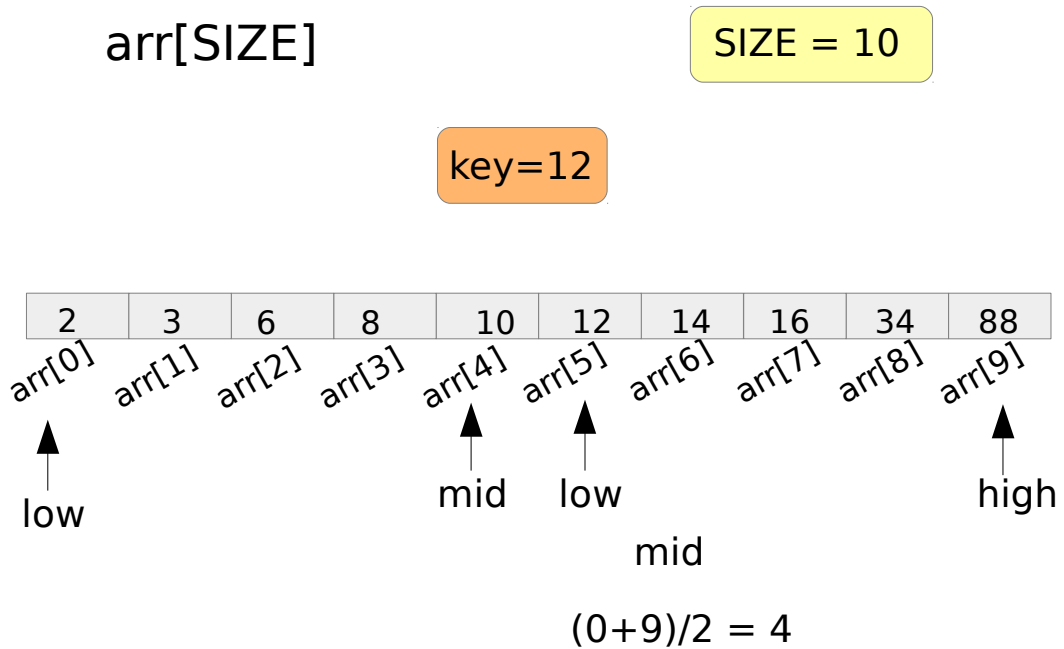
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



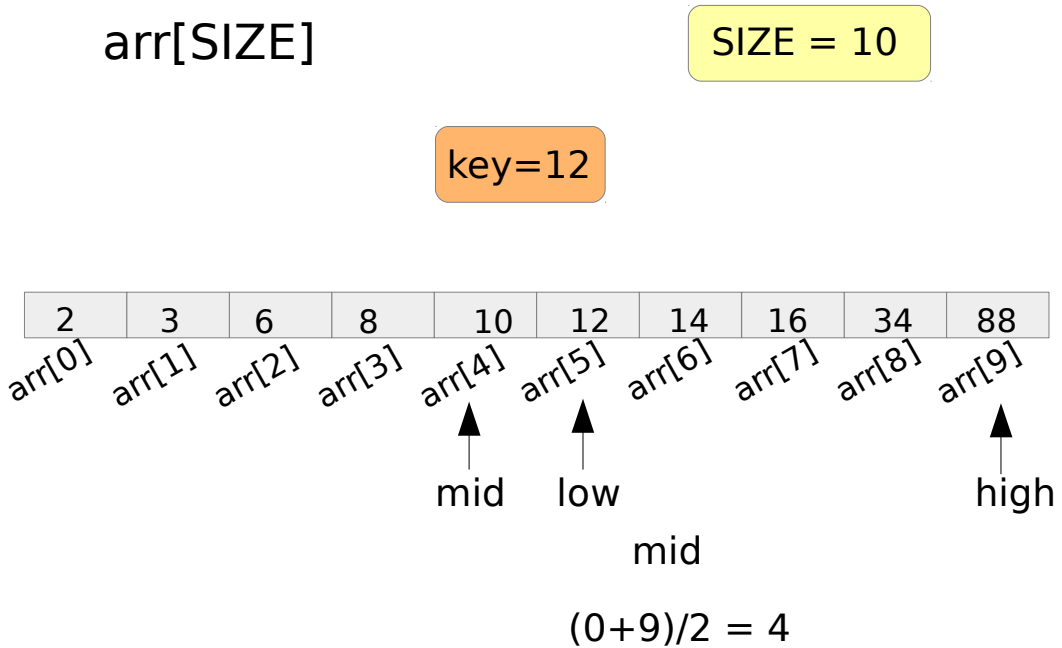
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



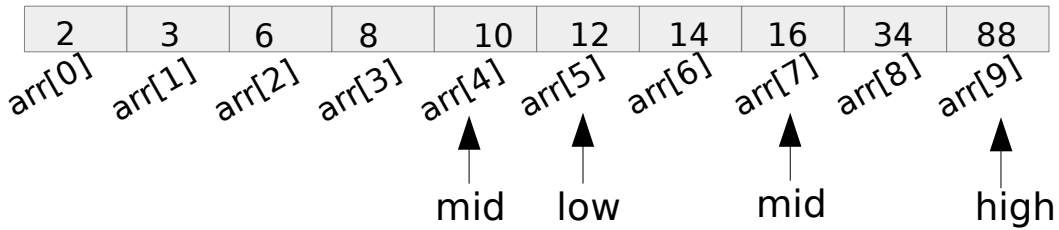
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



mid

(0+9)/2 = 4

(5+9)/2 = 7

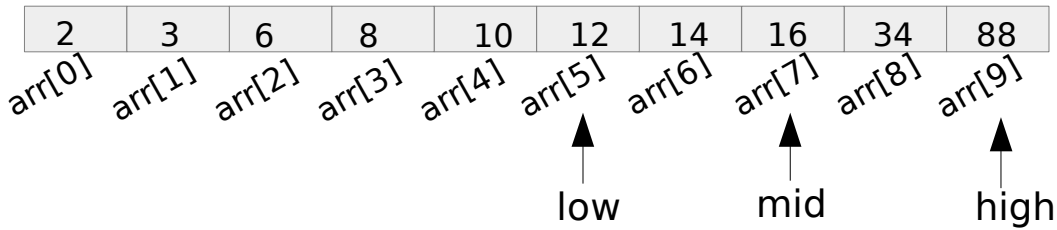
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



mid

(0+9)/2 = 4

(5+9)/2 = 7

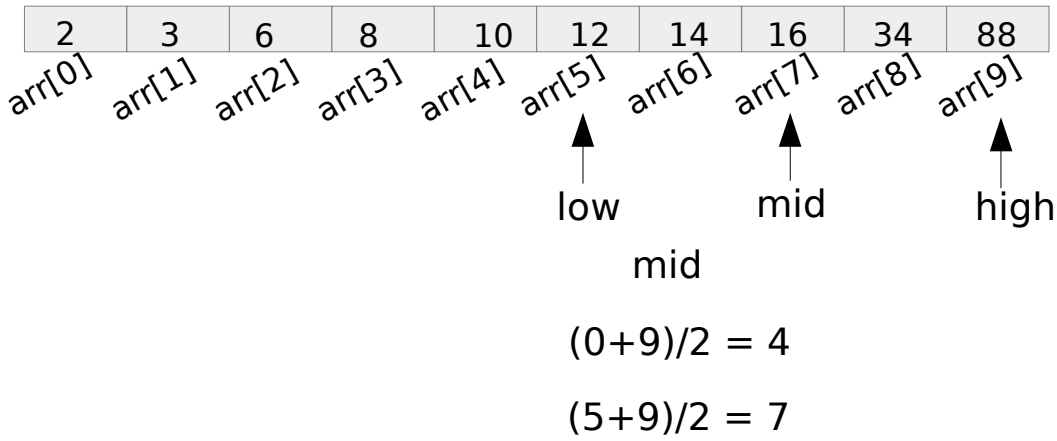
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



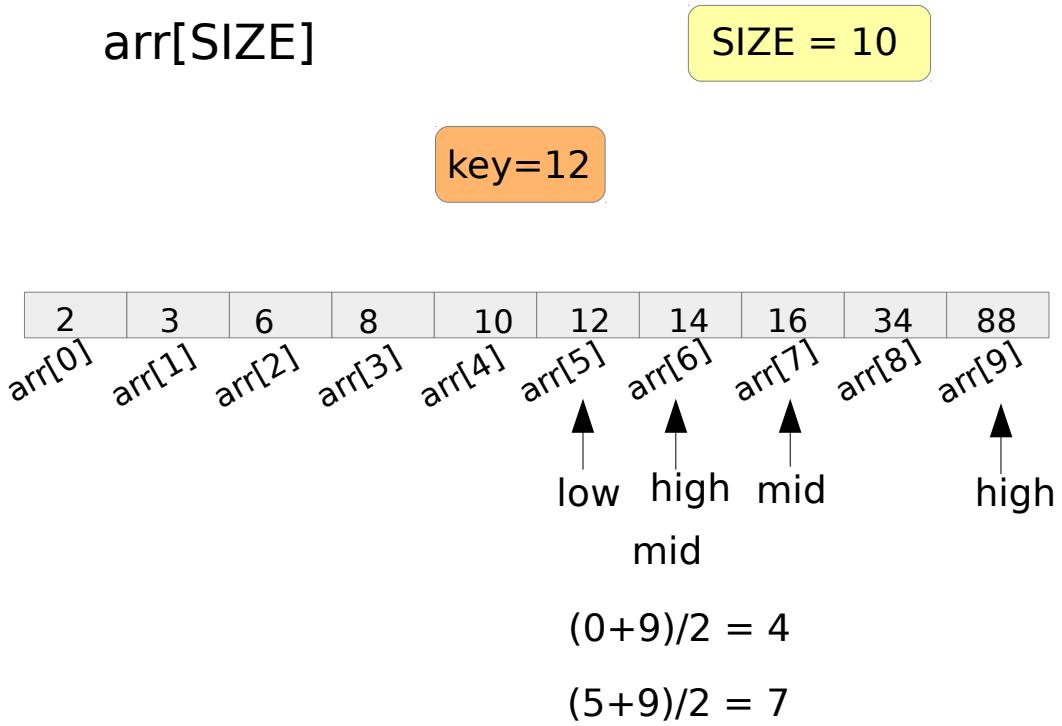
mid

(0+9)/2 = 4

(5+9)/2 = 7

Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



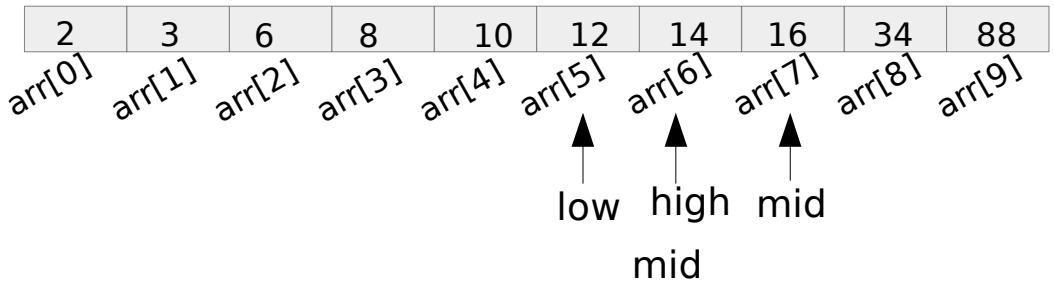
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



mid

(0+9)/2 = 4

(5+9)/2 = 7

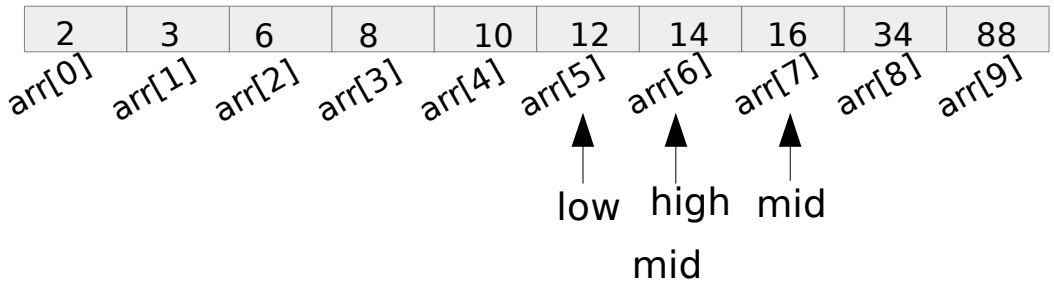
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



mid

(0+9)/2 = 4

(5+9)/2 = 7

(5+6)/2 = 5

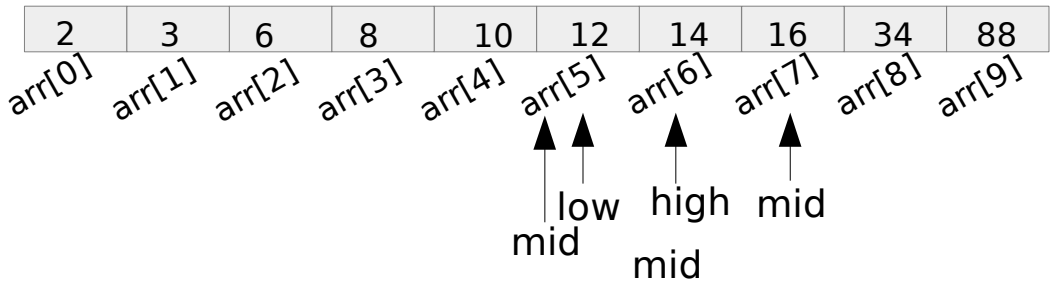
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



(0+9)/2 = 4

(5+9)/2 = 7

(5+6)/2 = 5

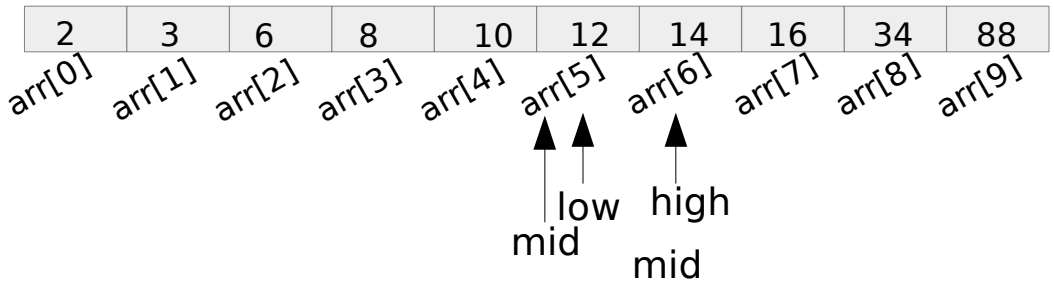
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=12



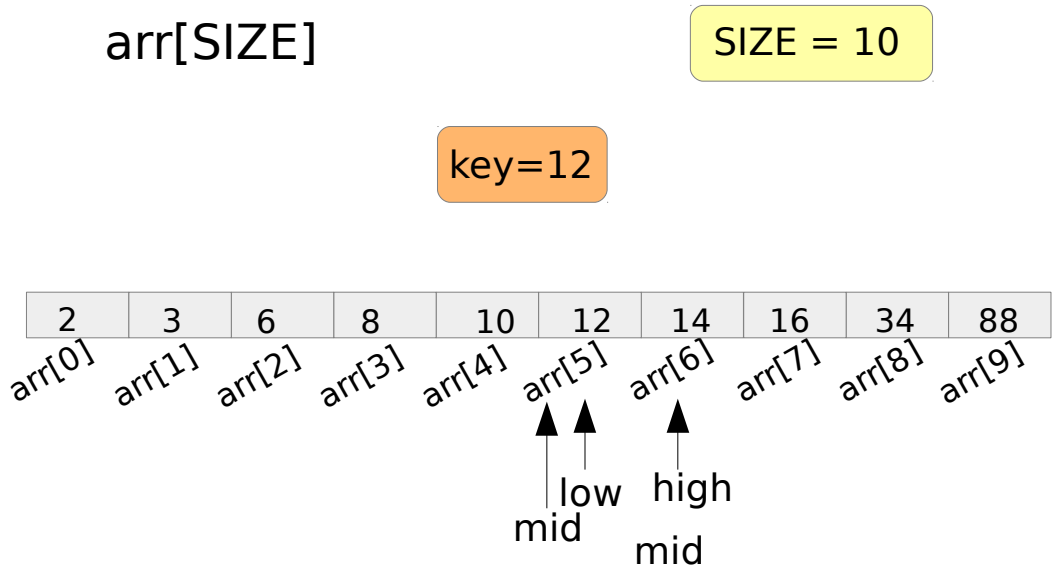
(0+9)/2 = 4

(5+9)/2 = 7

(5+6)/2 = 5

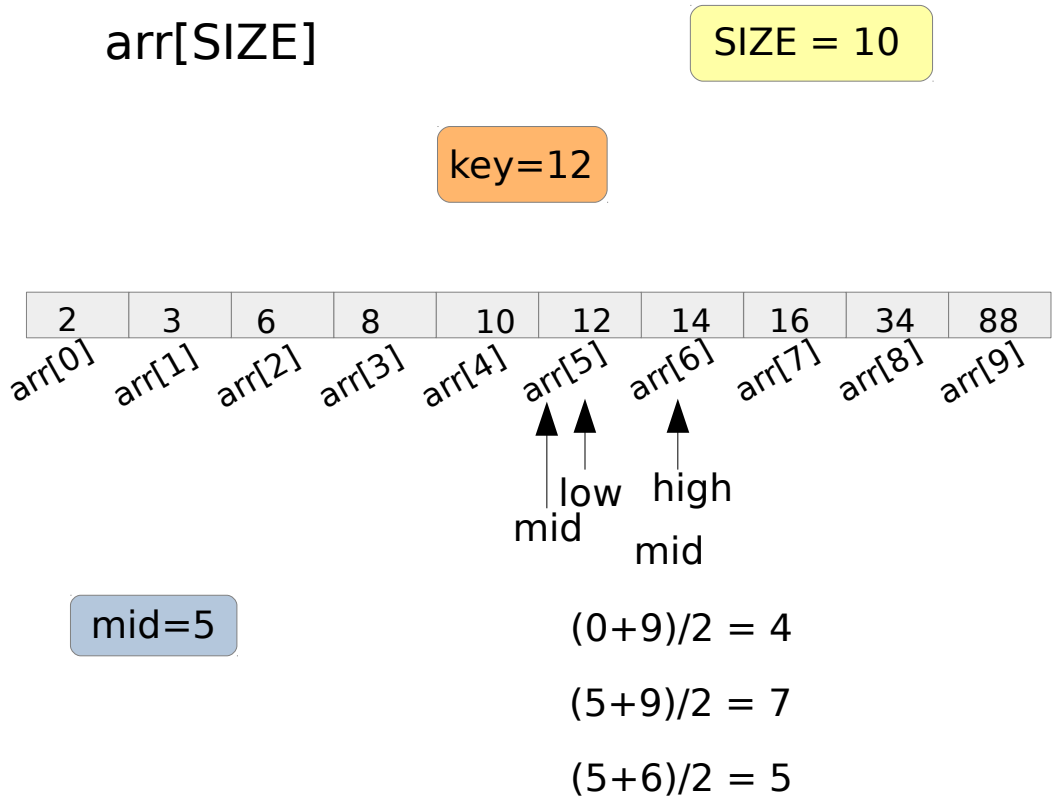
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



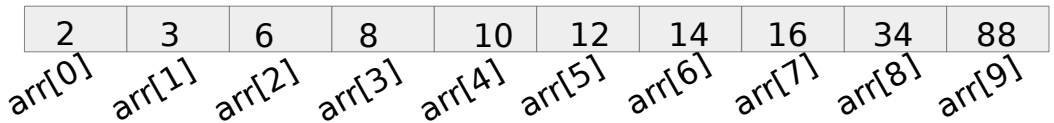
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

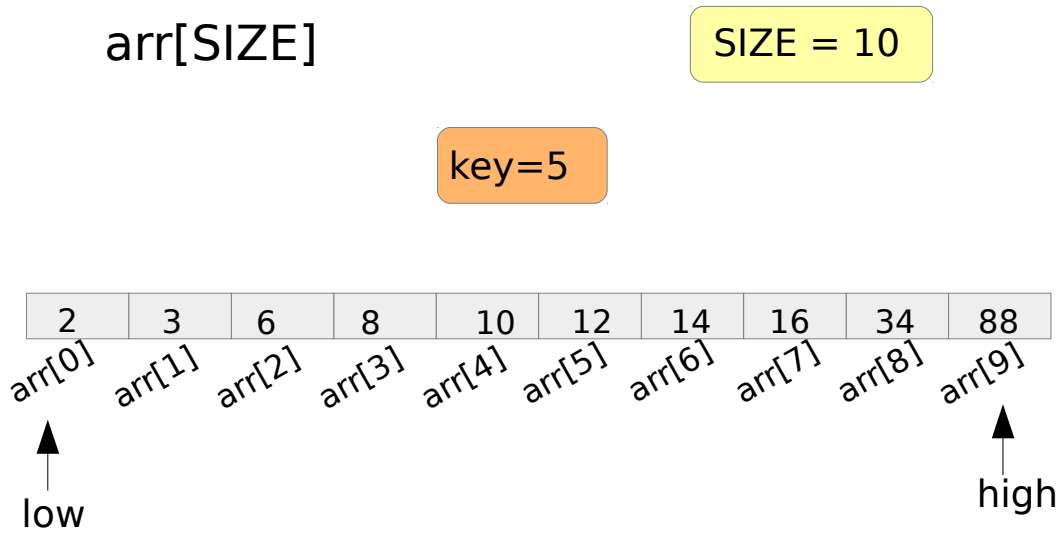
SIZE = 10

key=5



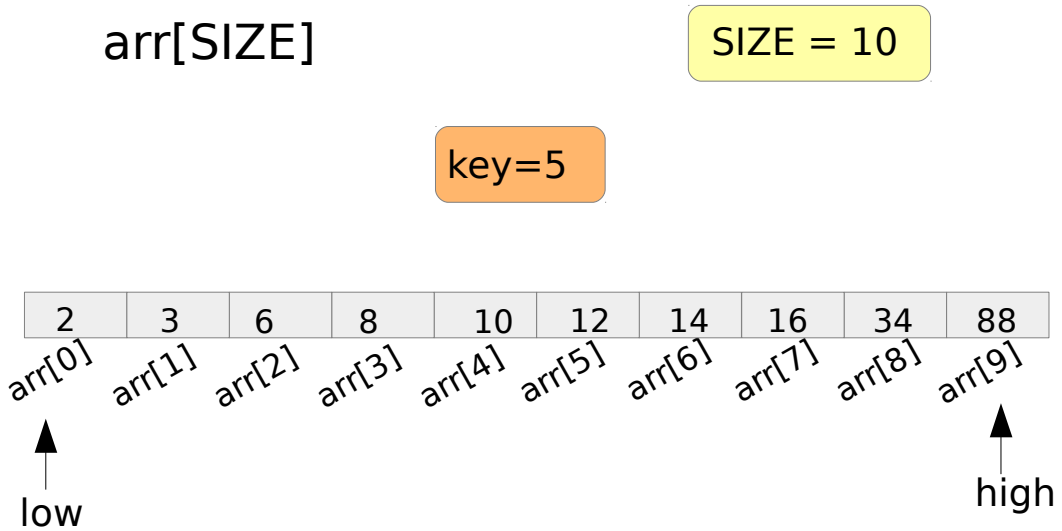
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



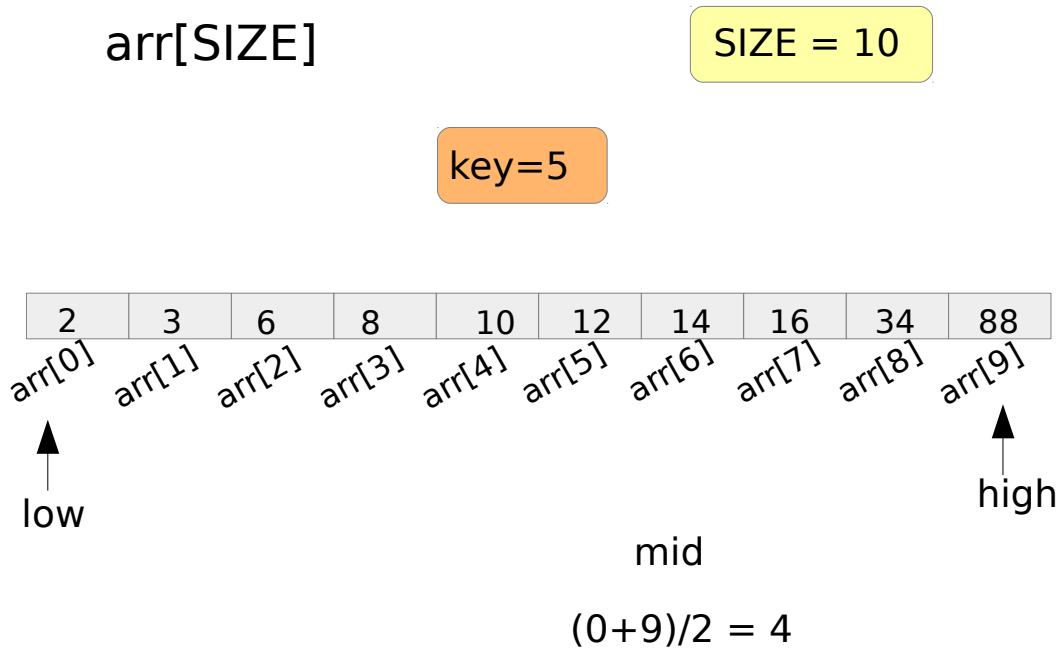
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



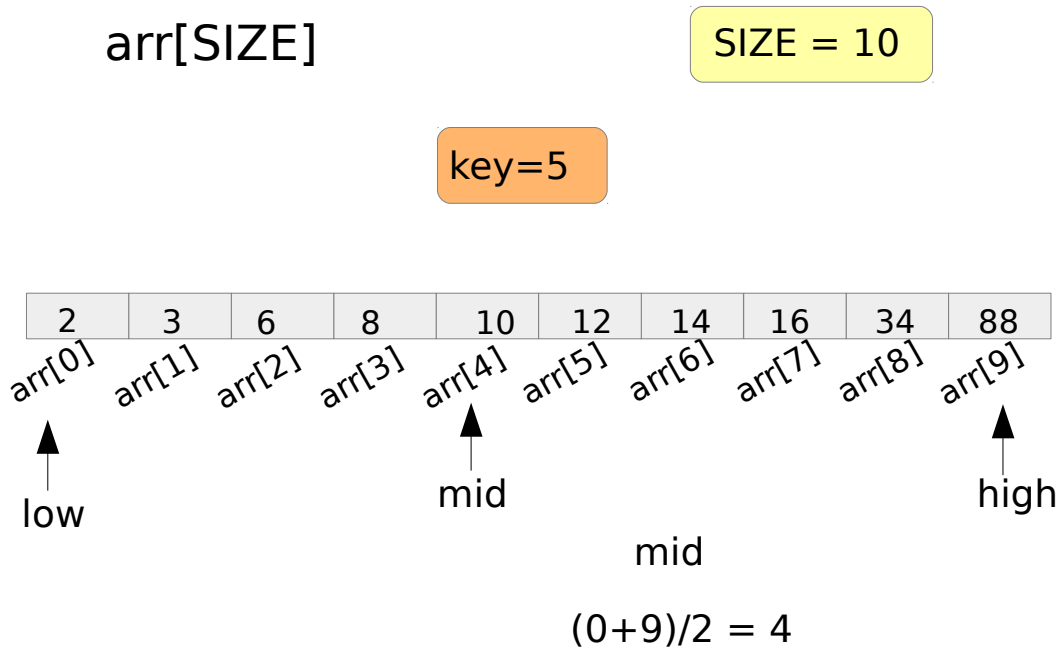
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```



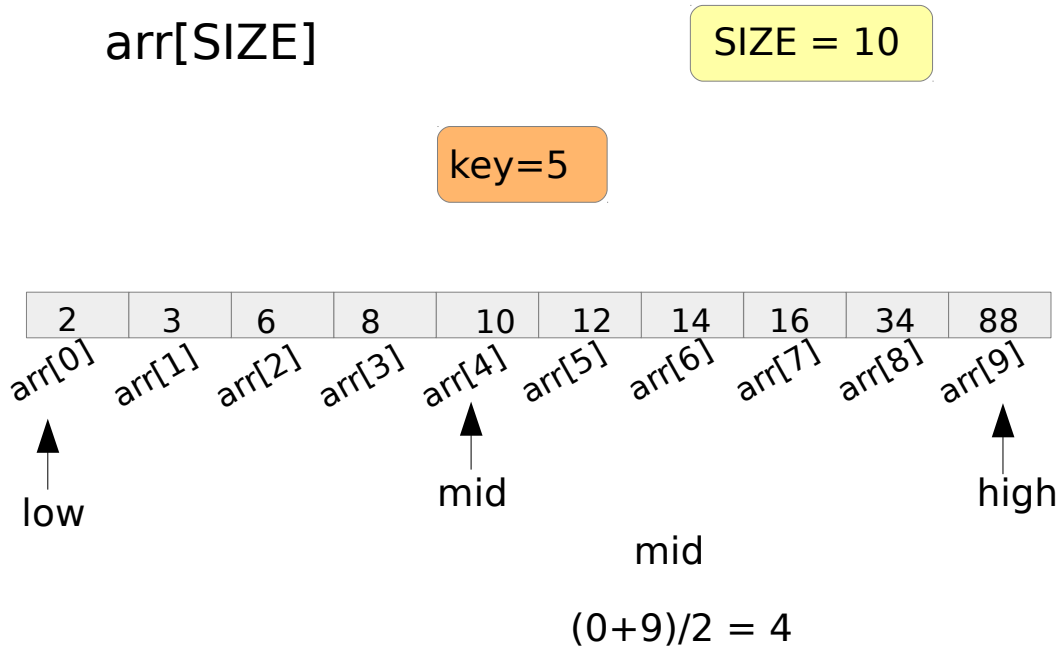
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```



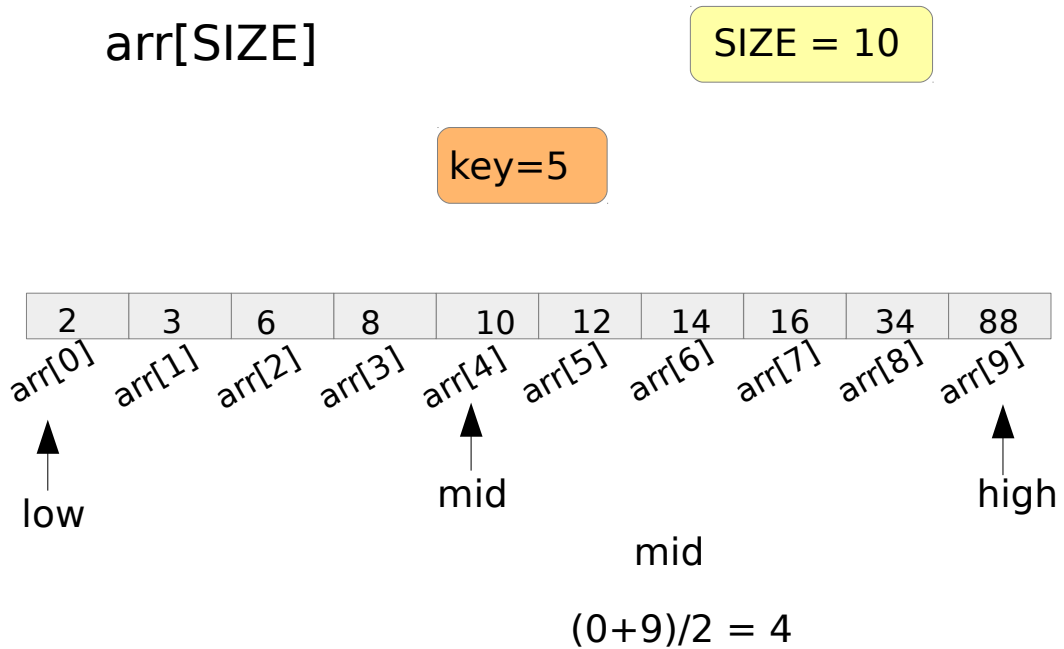
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



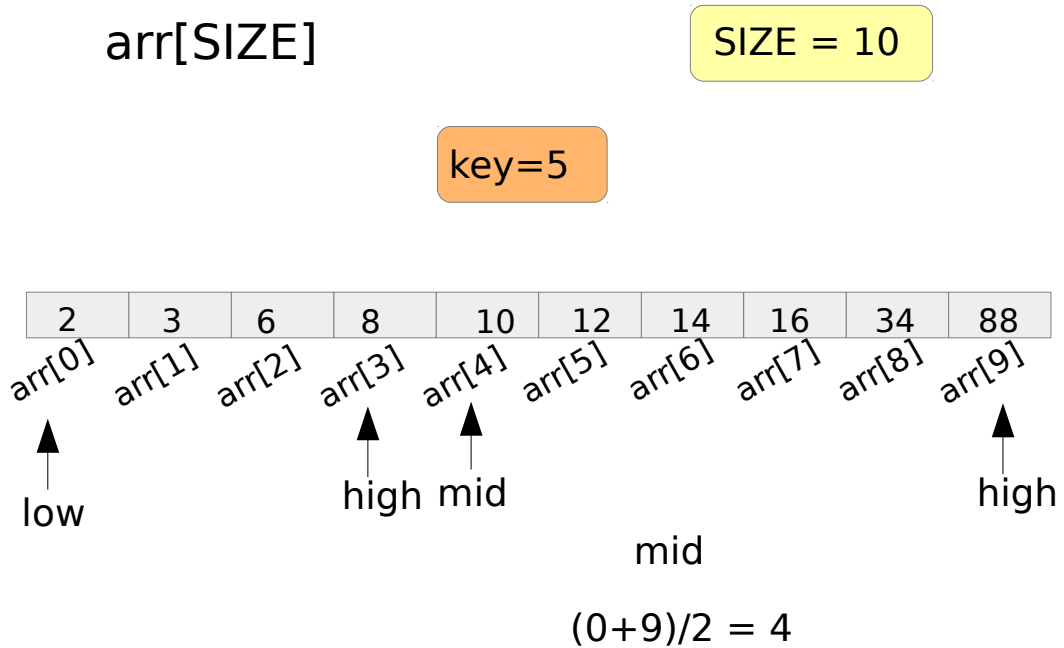
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



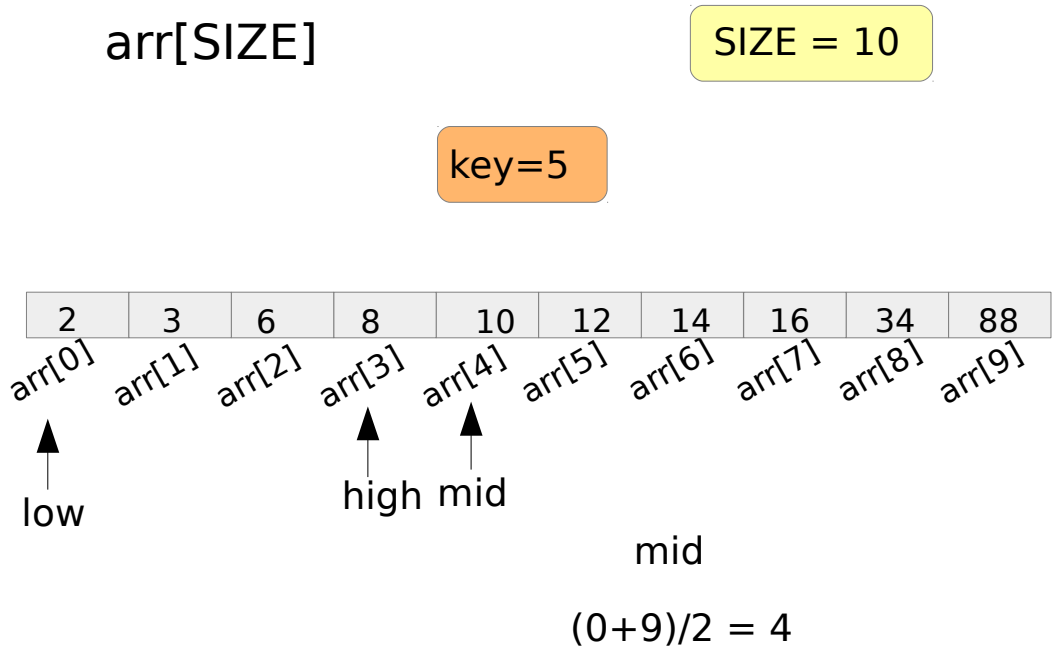
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



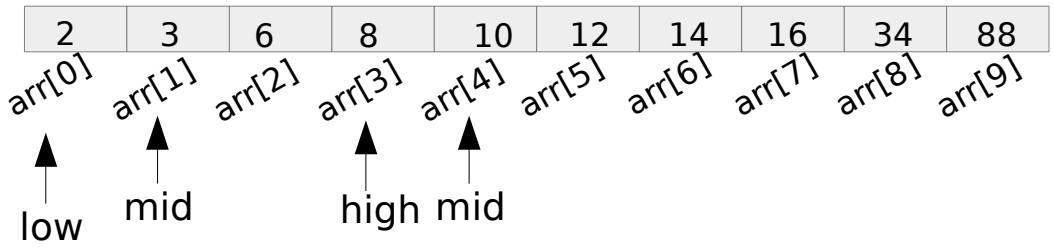
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
     if (arr[mid] == key )
          return mid
     else if (key < a[mid])
          high = mid - 1
     else
          low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



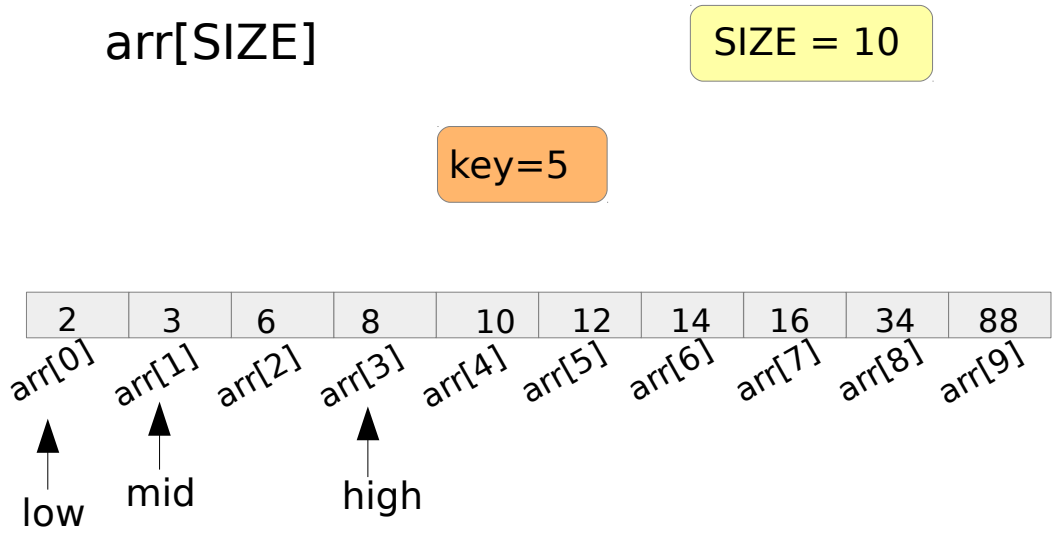
mid

(0+9)/2 = 4

(0+3)/2 = 1

Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



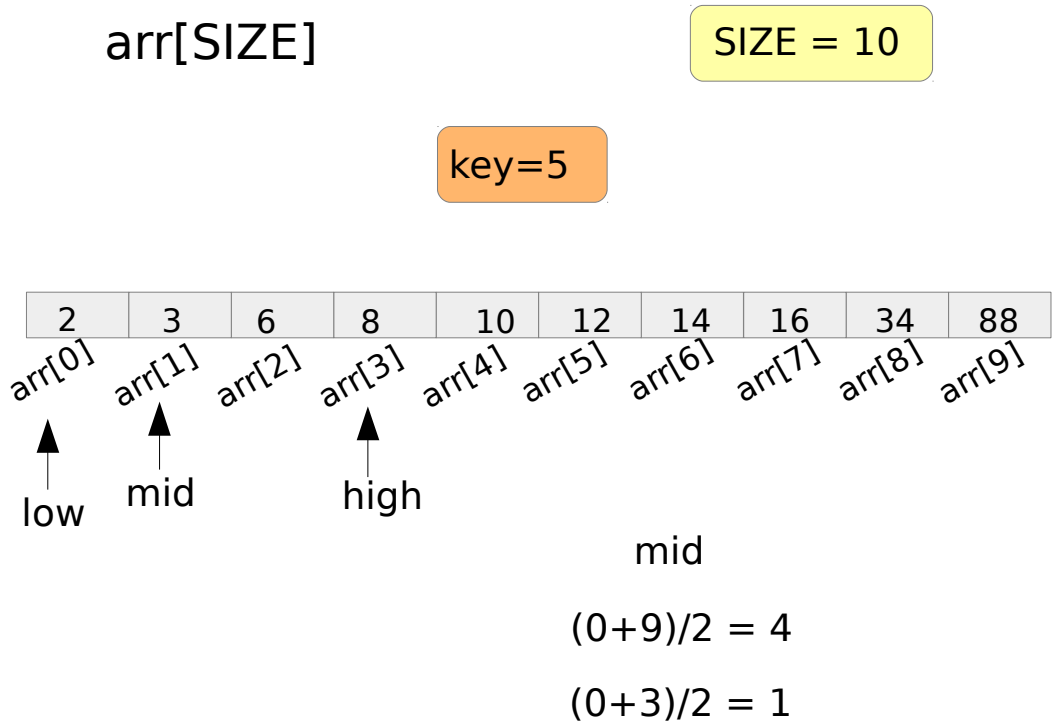
mid

(0+9)/2 = 4

(0+3)/2 = 1

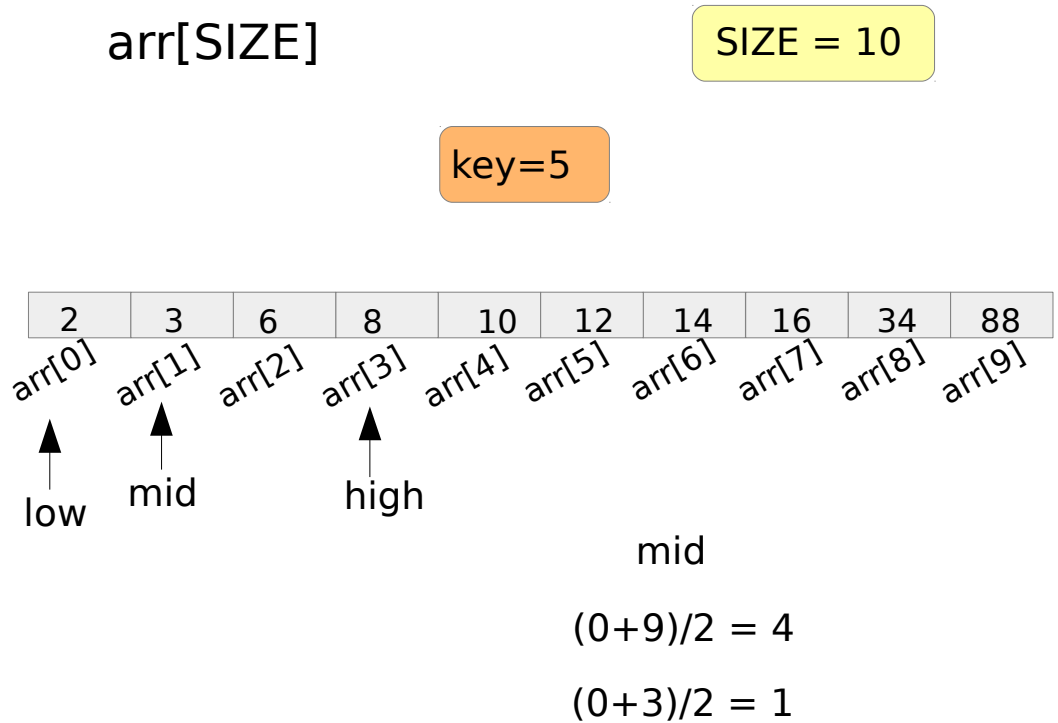
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



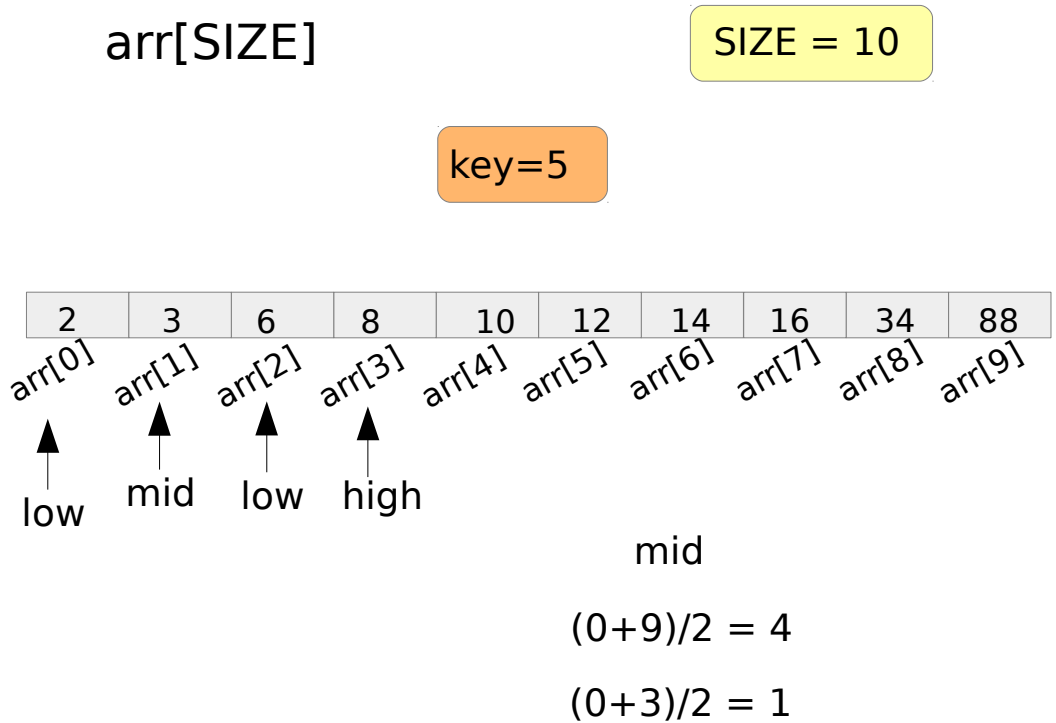
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



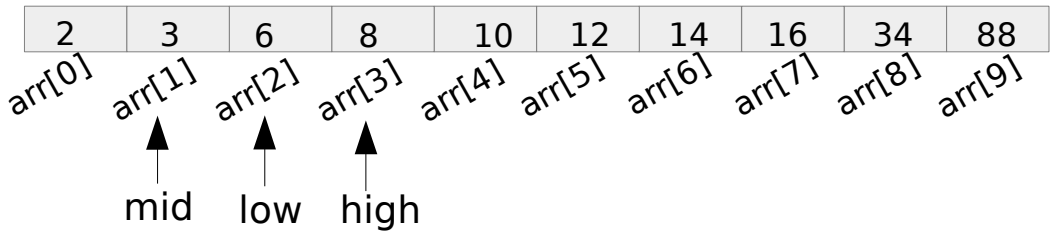
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



mid

(0+9)/2 = 4

(0+3)/2 = 1

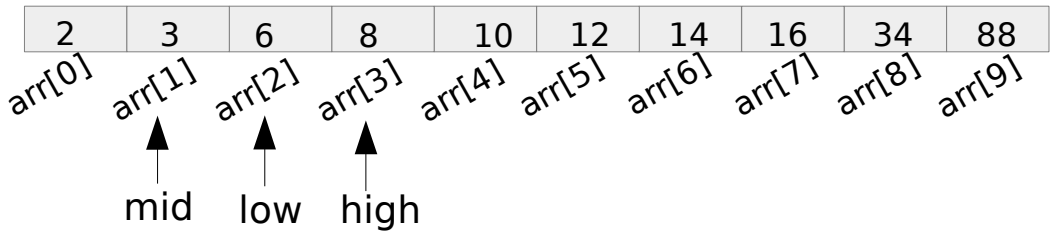
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
      if (arr[mid] == key )
            return mid
      else if (key < a[mid])
            high = mid - 1
      else
            low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



mid

(0+9)/2 = 4

(0+3)/2 = 1

(2+3)/2 = 2

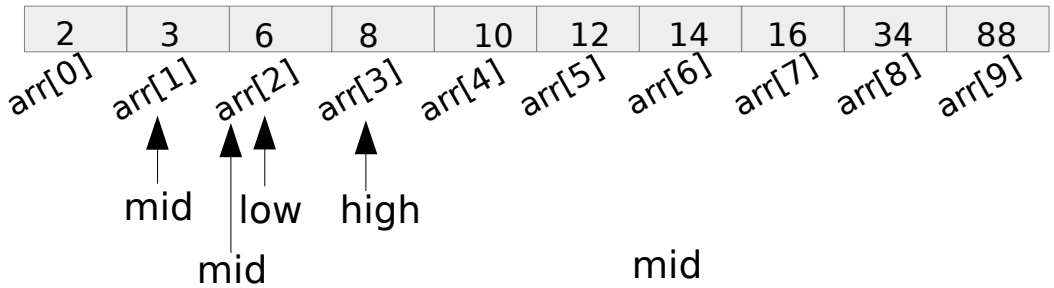
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
      if (arr[mid] == key )
            return mid
      else if (key < a[mid])
            high = mid - 1
      else
            low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



mid

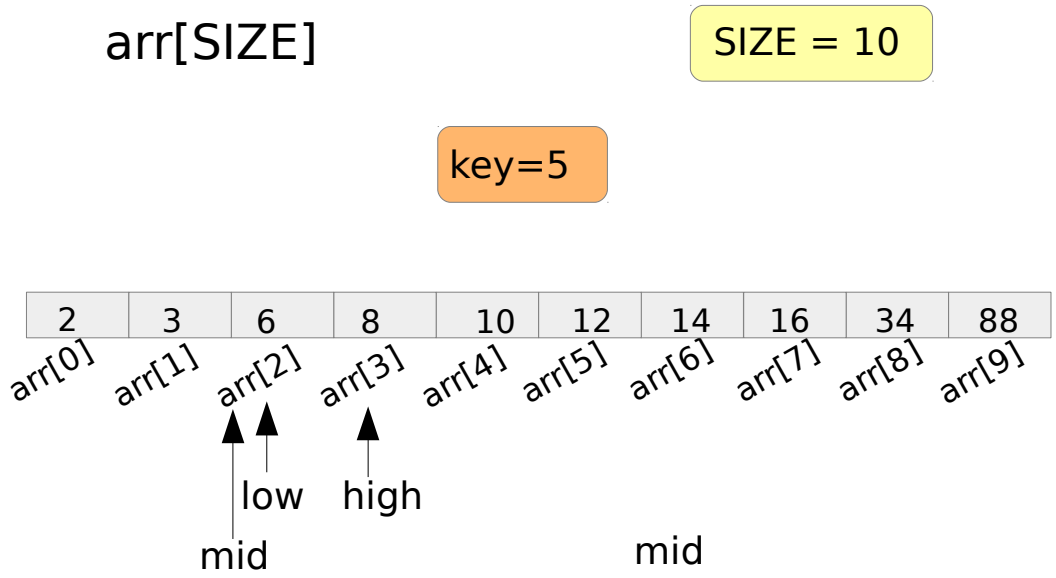
(0+9)/2 = 4

(0+3)/2 = 1

(2+3)/2 = 2

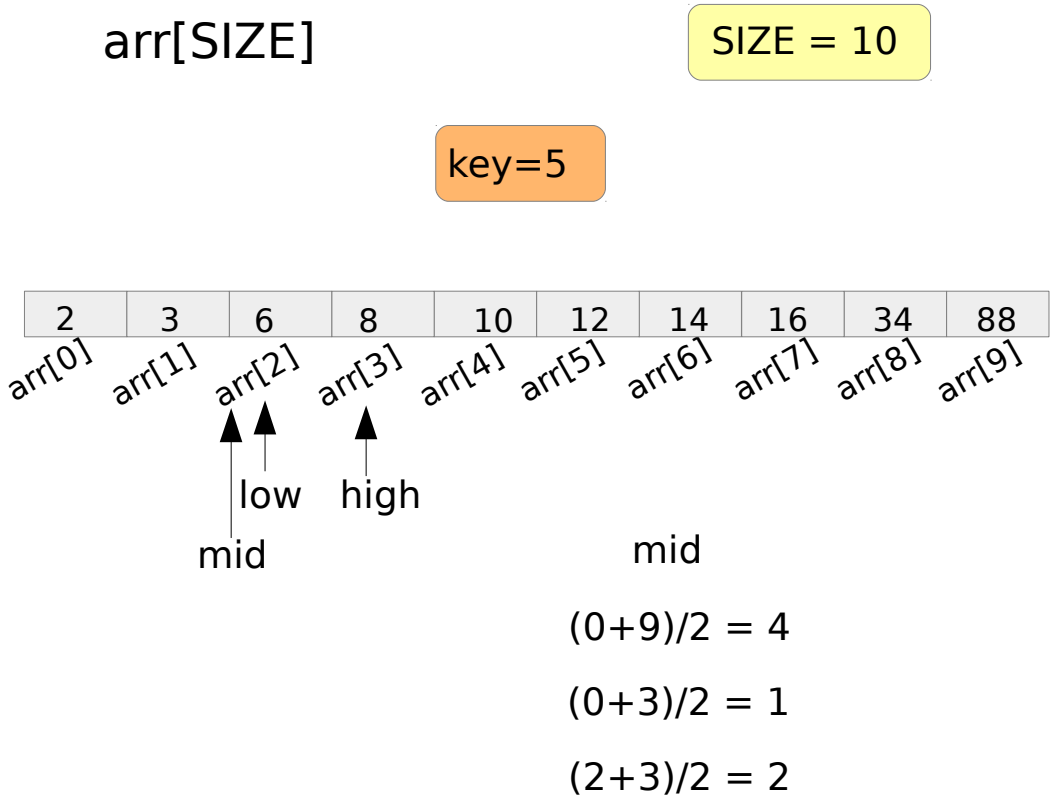
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```



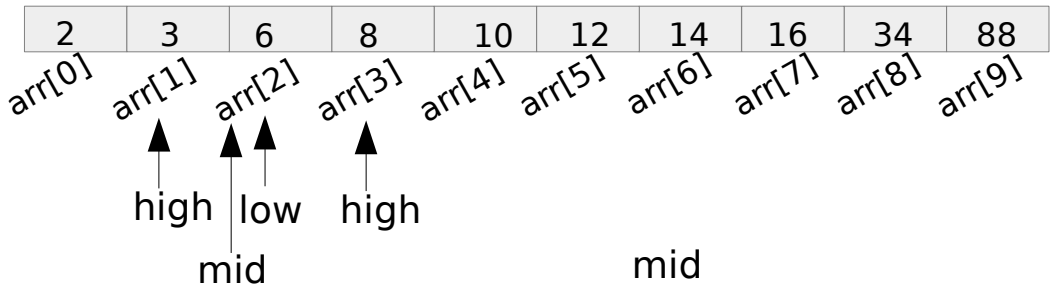
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



mid

(0+9)/2 = 4

(0+3)/2 = 1

(2+3)/2 = 2

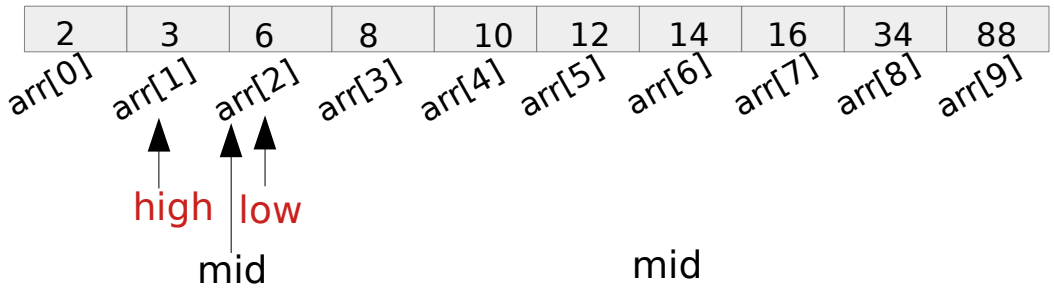
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



mid

(0+9)/2 = 4

(0+3)/2 = 1

(2+3)/2 = 2

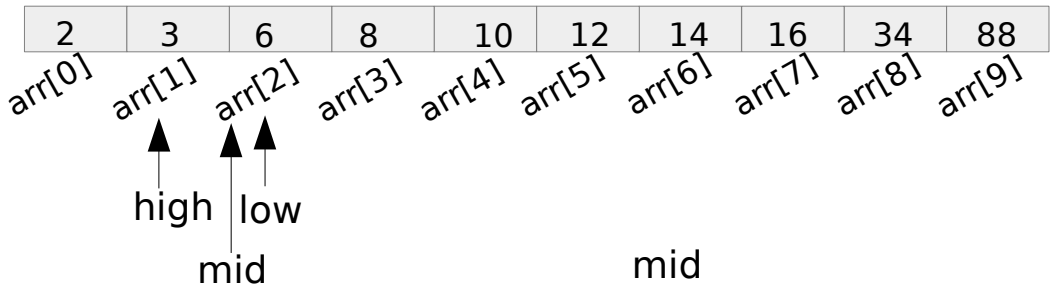
Binary_search_iterative(arr,key)

```
low = 0  ,high = size-1
while ( low <= high )
    mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
    else if (key < a[mid])
        high = mid - 1
    else
        low = mid + 1
return -1
```

arr[SIZE]

SIZE = 10

key=5



mid

(0+9)/2 = 4

(0+3)/2 = 1

(2+3)/2 = 2

CODE FOR THINGS

Code - Binary Search Iterative