Function A is an implementation of binary search algorithm to find the largest index of an integer **k** in a non_descending size-n array **Arr**. If k does not exist in **Arr**, return -1. But there are bugs in the code, please find them and fix them.

```
1. int A(int Arr[], int k)
2. {
3.     int min = 0, max = Arr.length-1, mid;
4.     while(min < max){
5.         mid =min+ (max - min)/2;
6.         if (Arr[mid] < k){
7.             min = mid;
8.         }else{
9.             max = mid – 1;
10.        }
11.    }
12.    if(Arr[max] == k){
13.        return max;
14.    }else{
15.        return -1;
16.    }
17.}
```

Function A is an implementation of binary search algorithm to find the largest index of an integer **k** in a non_descending size-n array **Arr**. If k does not exist in **Arr**, return -1. But there are bugs in the code, please find them and fix them.

```
1. int A(int Arr[], int k)
2. {
3.     int min = 0, max = Arr.length-1, mid;
4.     while(min < max){
5.         mid =min+ (max - min)/2;
6.         if (Arr[mid] < k){
7.             min = mid+1;
8.         }else{
9.             max = mid;
10.        }
11.    }
12.    if(Arr[max] == k){
13.        return max;
14.    }else{
15.        return -1;
16.    }
17.}
```

```
1. int A(int Arr[], int v)
2. {
3.     int min = 0, max = Arr.length, mid;
4.     while(min < max){
5.         mid = (max + min)/2;
6.         if (mid < k){
7.             min = mid;
8.         }else{
9.             max = mid – 1;
10.        }
11.    }
12.    if(Arr[max] == k){
13.        return max;
14.    }else{
15.        return -1;
16.    }
17.}
```

Wrong answer, try array 1,2,2 and find 2

Function A is an implementation of binary search algorithm to find the largest index of an integer **k** in a non_descending size-n array **Arr**. If k does not exist in **Arr**, return -1. But there are bugs in the code, please find them and fix them.

```
1. int A(int Arr[], int k)
2. {
3.     int min = 0, max = Arr.length-1, mid;
4.     while(min < max){
5.         mid =min+ (max – min)/2;
6.         if (Arr[mid] <= k){
7.             min = mid;
8.         }else{
9.             max = mid – 1;
10.        }
11.    }
12.    if(Arr[max] == k){
13.        return max;
14.    }else{
15.        return -1;
16.    }
17.}
```

```
1. int A(int Arr[], int v)
2. {
3.     int min = 0, max = Arr.length, mid;
4.     while(min < max){
5.         mid = (max + min)/2;
6.         if (mid < k){
7.             min = mid;
8.         }else{
9.             max = mid – 1;
10.        }
11.    }
12.    if(Arr[max] == k){
13.        return max;
14.    }else{
15.        return -1;
16.    }
17.}
```

Wrong answer, try array 1,2,2 and find 2

Function A is an implementation of binary search algorithm to find the largest index of an integer **k** in a non_descending size-n array **Arr**. If k does not exist in **Arr**, return -1. But there are bugs in the code, please find them and fix them.

```
1. int A(int Arr[], int k)
2. {
3.     int min = 0, max = Arr.length-1, mid;
4.     while(min < max){
5.         mid =min+ (max – min+1)/2;
6.         if (Arr[mid] <= k){
7.             min = mid;
8.         }else{
9.             max = mid – 1;
10.        }
11.    }
12.    if(Arr[max] == k){
13.        return max;
14.    }else{
15.        return -1;
16.    }
17.}
```

Correct answer