

Задача 4Е. Двоичный поиск

binsearch.in	binsearch.out
5 1 1 2 2 2 3 1 2 3	

```

int BSearchLeft(int arr[], int le, int ri, int
x){
    int result = -1;
    while (le <= ri){
        int mid = (le + ri) / 2;
        if (arr[mid] < x){
            le = mid + 1;
        }
        else if (arr[mid] > x){
            ri = mid - 1;
        }
        else{
            result = mid;
            ri = mid - 1;
        }
    }
    return result;
}

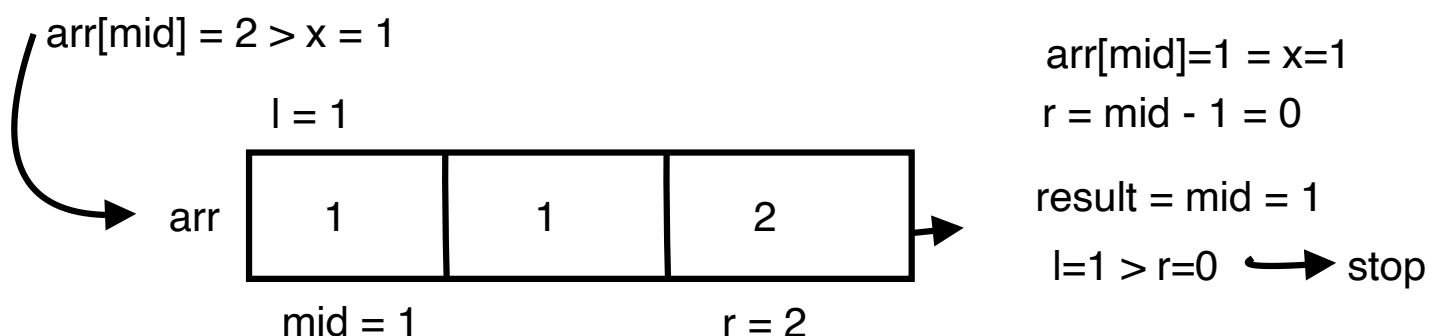
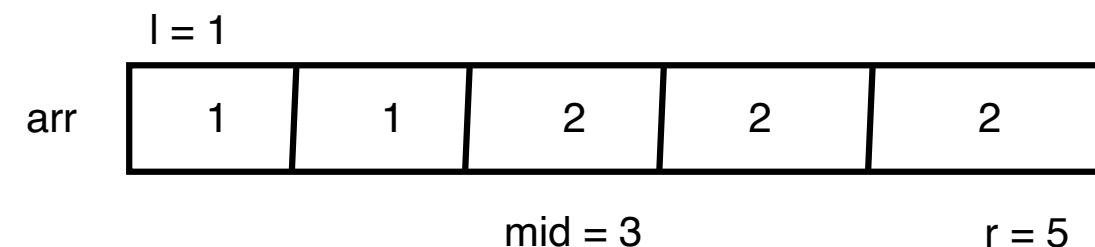
```

```

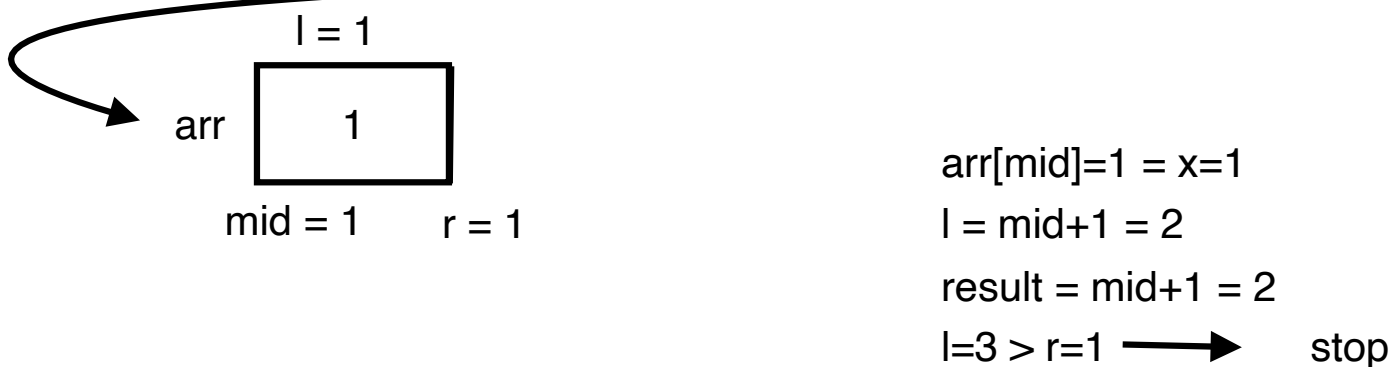
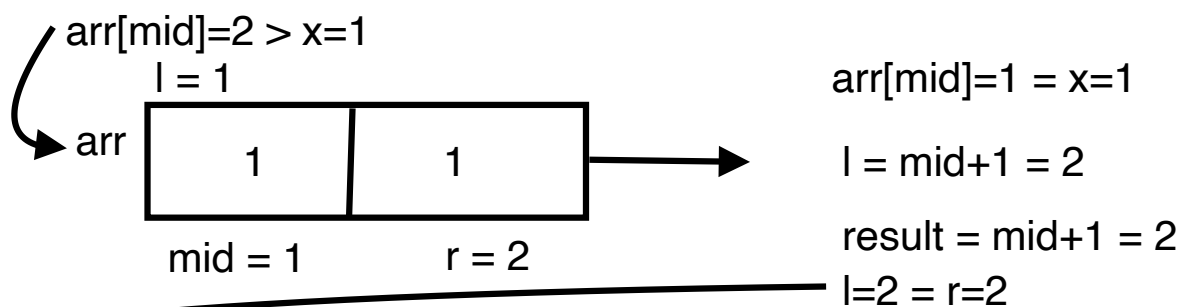
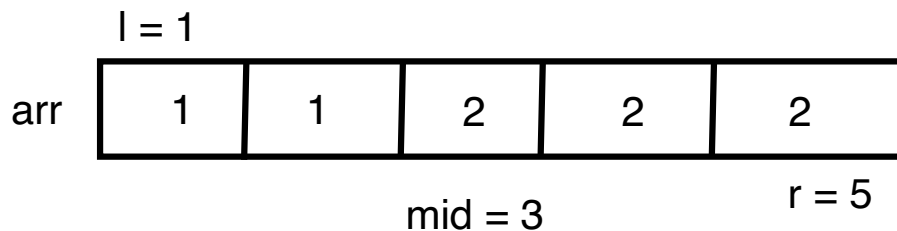
int BSearchRight(int arr[], int le, int ri,
int x){
    int result = -1;
    while (le <= ri) {
        int mid = (le + ri) / 2;
        if (arr[mid] < x){
            le = mid + 1;
        }
        else if (arr[mid] > x){
            ri = mid - 1;
        }
        else{
            result = mid+1;
            le = mid+1;
        }
    }
    return result;
}

```

1. номер первого вхождения 1



2. номер последнего вхождения 1



Для 2, 3 делаем то же самое.