1)

```
def mySqrt(x):
        if x == 0 or x == 1:
        return x

        left, right = 0, x
        while left <= right:
        mid = (left + right) // 2
        square = mid * mid

        if square == x:
        return mid
        elif square > x:
        right = mid - 1
        else:
        left = mid + 1

        return right

print(mySqrt(4))  # Output: 2
print(mySqrt(8))  # Output: 2
```

2)

```
def findPeakElement(nums):
        left = 0
        right = len(nums) - 1

        while left < right:
        mid = left + (right - left) // 2

        if nums[mid] < nums[mid + 1]:
        left = mid + 1
        else:
        right = mid

        return left

# Test the function
nums = [1, 2, 3, 1]
print(findPeakElement(nums))  # Output: 2

nums = [1, 2, 1, 3, 5, 6, 4]
print(findPeakElement(nums))  # Output: 5
```

3)

```
def missingNumber(nums):
        missing = len(nums)
```

```
        for i, num in enumerate(nums):
        missing ^= i ^ num
        return missing

# Test the function
nums = [3, 0, 1]
print(missingNumber(nums))  # Output: 2

nums = [0, 1]
print(missingNumber(nums))  # Output: 2

nums = [9, 6, 4, 2, 3, 5, 7, 0, 1]
print(missingNumber(nums))  # Output: 8

4)
def findDuplicate(nums):
        slow = nums[0]
        fast = nums[0]

        # Detect the cycle
        while True:
        slow = nums[slow]
        fast = nums[nums[fast]]
        if slow == fast:
        break

        # Find the start of the cycle
        slow = nums[0]
        while slow != fast:
        slow = nums[slow]
        fast = nums[fast]

        return slow

# Test the function
nums = [1, 3, 4, 2, 2]
print(findDuplicate(nums))  # Output: 2

nums = [3, 1, 3, 4, 2]
print(findDuplicate(nums))  # Output: 3

5)
def intersection(nums1, nums2):
        set1 = set(nums1)
        set2 = set(nums2)
        return list(set1.intersection(set2))

# Test the function
```

```
nums1 = [1, 2, 2, 1]
nums2 = [2, 2]
print(intersection(nums1, nums2))  # Output: [2]

nums1 = [4, 9, 5]
nums2 = [9, 4, 9, 8, 4]
print(intersection(nums1, nums2))  # Output: [9, 4]
```

6)

```
def findMin(nums):
        left = 0
        right = len(nums) - 1

        while left < right:
        mid = left + (right - left) // 2

        if nums[mid] > nums[right]:
        left = mid + 1
        else:
        right = mid

        return nums[left]

# Test the function
nums = [3, 4, 5, 1, 2]
print(findMin(nums))  # Output: 1

nums = [4, 5, 6, 7, 0, 1, 2]
print(findMin(nums))  # Output: 0

nums = [11, 13, 15, 17]
print(findMin(nums))  # Output: 11
```

7)

```
def searchRange(nums, target):
        left = findLeftPosition(nums, target)
        right = findRightPosition(nums, target)
        return [left, right]

def findLeftPosition(nums, target):
        left = 0
        right = len(nums) - 1
        result = -1

        while left <= right:
        mid = left + (right - left) // 2

        if nums[mid] == target:
```

```
            result = mid
            right = mid - 1
            elif nums[mid] < target:
            left = mid + 1
            else:
            right = mid - 1

            return result

def findRightPosition(nums, target):
            left = 0
            right = len(nums) - 1
            result = -1

            while left <= right:
            mid = left + (right - left) // 2

            if nums[mid] == target:
            result = mid
            left = mid + 1
            elif nums[mid] < target:
            left = mid + 1
            else:
            right = mid - 1

            return result

# Test the function
nums = [5, 7, 7, 8, 8, 10]
target = 8
print(searchRange(nums, target))  # Output: [3, 4]

nums = [5, 7, 7, 8, 8, 10]
target = 6
print(searchRange(nums, target))  # Output: [-1, -1]

nums = []
target = 0
print(searchRange(nums, target))  # Output: [-1, -1]

8)
from collections import Counter

def intersect(nums1, nums2):
            counter1 = Counter(nums1)
            result = []

            for num in nums2:
```

```
        if num in counter1 and counter1[num] > 0:
        result.append(num)
        counter1[num] -= 1

        return result

# Test the function
nums1 = [1, 2, 2, 1]
nums2 = [2, 2]
print(intersect(nums1, nums2))  # Output: [2, 2]

nums1 = [4, 9, 5]
nums2 = [9, 4, 9, 8, 4]
print(intersect(nums1, nums2))  # Output: [4, 9]
```