

# Задачи върху алгоритми за търсене

## 1 Linear Search:

### 1.1 Основна задача с Linear Search

Напишете функция, която използва линейно търсене, за да намери първия индекс на дадено число в масив. Ако числото не е намерено, функцията трябва да върне -1.

Пример:

Вход: масив [2, 5, 7, 10, 15] и число 7

Изход: 2

### 1.2 Търсене на най-голямото и най-малкото число

Използвайте линейно търсене, за да намерите най-голямото и най-малкото число в масив.

Пример:

Вход: [3, 6, 1, 8, 4]

Изход: Най-малко = 1, Най-голямо = 8

### 1.3 Брой на срещанията на елемент

Напишете функция, която брой колко пъти дадено число се среща в масив, използвайки линейно търсене.

Пример:

Вход: масив [5, 3, 5, 2, 5] и число 5

Изход: 3

## 2 Binary Search:

### 2.1 Основна задача с Binary Search

Напишете функция, която използва двоично търсене, за да провери дали дадено число съществува в сортиран масив. Върнете индекса на числото, ако е намерено, или -1, ако не е намерено.

Пример:

Вход: масив [1, 3, 5, 7, 9] и число 5

Изход: 2

### 2.2 Намиране на най-малко и най-голямо срещане на число

Даден е сортиран масив, съдържащ повторения на едно и също число. Напишете функция, която използва двоично търсене, за да намери най-малкия и най-големия индекс на дадено число.

Пример:

Вход: масив [2, 4, 4, 4, 8, 10] и число 4

Изход: Първи индекс = 1, Последен индекс = 3

### 2.3 Търсене на квадратен корен чрез Binary Search

Напишете функция, която използва двоично търсене, за да намери целочисления квадратен корен на дадено число.

Пример:

Вход: 25

Изход: 5

## 3 Skip Search:

### 3.1 Основна задача със Skip Search

Даден е сортиран масив, който съдържа голям брой елементи. Реализирайте Skip Search, за да намерите индекса на даден елемент. Ако елементът не съществува, върнете -1.

Пример:

Вход: масив [1, 2, 5, 7, 9, 15, 20, 21, 25, 30] и число 15

Изход: 5

### 3.2 Търсене на стойност с ограничени стъпки

Даден е сортиран масив и целеви елемент. Напишете алгоритъм, използващ Skip Search, който да ограничи броя на проверките. Ако стойността бъде намерена преди достигане на максималния брой стъпки, върнете индекса, иначе върнете -1.

Пример: Вход: масив [1, 3, 6, 8, 10, 14, 18, 20, 22] и число 14

Изход: 5

### 3.3 Оптимизирано търсене в дълъг списък

Даден е дълъг списък от сортирани числа, в който повечето елементи се повтарят. Използвайте Skip Search, за да намерите първия индекс на уникален елемент в списъка.

Пример:

Вход: [2, 2, 2, 5, 5, 9, 9, 9, 12, 15] и число 9

Изход: 5