

1. Напишете програма, която по даден масив намира индекса на последното срещане на дадено число.
2. Да се състави програма, която прочита буквите на парола с максимална дължина 10 символа и копира паролата в нов масив като слага цифрите от 0 до 9 последователно между всеки две букви, за да увеличи силата на паролата.
3. След всяко четно число в масив вмъкнете сумата на всички нечетни числа предхождащи го в масива.

*Вход:* 1 2 3 4 5

*Изход:* 1 2 1 3 4 2 5

4. Напишете функция, която сортира масив в низходящ ред.
5. Сортирайте елементите на масив по броя срещания на елемента. Елементите на масива са от  $[0, 100]$ .

*Вход:* 1 4 2 4 3 4 1 1 4 3

*Изход:* 4 4 4 4 1 1 1 3 3 2

6. Напишете програма, която по даден интервал от числа, познава намисленото от вас число, като вие казвате дали текущото предположение на компютъра е по-малко или по-голямо от вашето число.
7. Напишете програма, която намира най-дългата последователност в един масив от **цели числа**.

8. Напишете програма, която приема 2 сортирани масива от цели числа и ги слива в един, който е сортиран в низходящ ред.

9. Напишете програма, която по даден сортиран масив от числа намира дали дадено число е в масива, правейки възможно най-малко сравнения.

10. Напишете програма, която да трие елемент с индекс **k** от масив с **n** на брой положителни числа и да измества другите елементи, като в получената се дупка сложите числото -1, което бележи, че е невалидно поле. Входът се състои от числото **n**, последвано от **n** числа и накрая се въвежда **k**.

*Вход:*

5

3 7 23 5 1

2

*Изход*

3 7 5 1 -1

11. Напишете програма, която по даден масив и число намира дали има двойка числа, чиито сбор е даденото число.

12. Да се състави програма, която прочита **N** битовете на число в двоична бройна система и отпечата стойността в осмична бройна система. Входът се състои от последователност от 0 и 1 с дължина **N** ( $N \leq 128$ ) и разделени с интервали.

*Вход:*

10

0 1 0 0 0 0 1 0 1 0

*Изход:*

412

13. Напишете програма, която сортира масив от средата навън, така че в средата е най-големият елемент, а в краищата са най-малките. (Сумата на елементите в лявата половина е по-малка или равна на сумата на елементите в дясната половина).

*Вход:* 1 4 2 9 7 5 3 6 8

*Изход:* 1 3 5 7 9 8 6 4 2