**JavaScript LOOPS & ARRAYS**

1. Напишете JavaScript програма, която отпечатва в конзолата числата от 1 до 10.
2. Напишете JavaScript програма, която създава масив от 20 елемента, като стойността на всеки елемент се получава от индекса му, умножен по 5.
3. Напишете JavaScript програма, която намира елемента с максимална и елемента с минимална стойност и ги отпечатва в конзолата по показания начин. Не използвайте готови функции.

|  |  |
| --- | --- |
| **Input** | **Output** |
| [1, 2, 1, 15, 20, 5, 7, 31] | Min -> 1 |
| Max -> 31 |
| [2, 2, 2, 2, 2] | Min -> 2 |
| Max -> 2 |
| [500, 1, -23, 0, -300, 28, 35, 12] | Min -> -300 |
| Max -> 500 |

1. Напишете JavaScript програма, която по зададен масив от числа и число, отпечатва **bigger** или **not bigger**, в зависимост от това дали елементът с индекс, подаденото число е по-голям от съседните си два елемента. Ако елементът е в началото/края на масива – програмата ви отпечатва – **only one neighbour’** и **invalid index** в случай, че няма елемент с такъв индекс.

|  |  |
| --- | --- |
|  |  |
| **Input** | **Output** |
| 2, [1, 2,3, 3, 5] | not bigger |
| 2, [1, 2,5, 3, 4] | bigger |
| 5, [1, 2, 5, 3, 4] | invalid index |
| 0, [1, 2, 5, 3, 4] | only one neighbor |

1. Напишете JavaScript програма, която отпечатва в конзолата думите в стринг отзад напред, запазвайки последователността им.

|  |  |
| --- | --- |
| **Input** | **Output** |
| Hello, how are you.' | ,olleH woh era .uoy |
| Life is pretty good, isn’t it?' | efiL si ytterp,doog t'nsi ?ti |

1. Напишете JavaScript програма, която намира максималната последователност от еднакви числа в масив и я отпечатва в конзолата. /числата са подадени като стрингове/

|  |  |
| --- | --- |
| **Input** | **Output** |
| ['10', '2', '1', '1', '2', '3', '3', '2', '2', '2', '1'] | 3 |

1. Напишете JavaScript програма, която отпечатва матрица, като посочения пример, по зададено цяло число N.
   1. Постигнете същия ефект без да използвате вложени цикли.

## Input - Масив с един елемент – числото N.

|  |  |
| --- | --- |
| Input | Output |
| ['2'] | 1 2  2 3 |
|  |  |
| ['3'] | 1 2 3  2 3 4  3 4 5 |
|  |  |
| ['4'] | 1 2 3 4 2 3 4 5  3 4 5 6  4 5 6 7 |

# 16 към 10. Като използвате цикли, напишете javascript програма, която конвертира шестнайсетични към десетични числа. Входни данни – масив с един елемент – шестнайсетично число

|  |  |
| --- | --- |
| Input | Output |
| ['FE'] | 254 |
| ['1AE3'] | 6883 |
| ['4ED528CBB4'] | 338583669684 |
|  |  |

1. Отпечатайте всички четирицифрени числа **ABCD**, за които **A**+**B** = **C**+**D** (известни като щастливи числа).
2. Напишете JavaScript програма, която намира най-често повтарящата се дума в текст и я отпечатва в конзолата, както и колко пъти се повтаря във формата "дума-> брой". Ако няколко думи са с максимален брой повторения – отпечатайте ги в азбучен ред.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 'in the middle of the night' | the -> 2 times |
| 'Welcome to SoftUni. Welcome to Java. Welcome everyone.' | welcome -> 3 times |
| 'Hello my friend, hello my darling. Come on, come here. Welcome, welcome darling.' | come -> 2 times |
| darling -> 2 times |
| hello -> 2 times |
| my -> 2 times |
| welcome -> 2 times |

# Selection sort

Да сортираме масив, означава да подредим елементите му в нарастващ ред. Напишете програма за сортиране на масив. Използвате [Selection sort](http://en.wikipedia.org/wiki/Selection_sort) – намираме най-малкия елемент, поставяме го на първо място, намираме най-малкия елемент от оставащите – преместваме го на второ място и т. н.

1. Иво и Ники трябва да присъстват на конференция, но закъсняват много. Нямат какво да правят и решават да запълнят времето си като броят гумите на превозните средства, намиращи се на паркинга на хотела. Общият брой на всички гуми означават с **S**. На паркинга има три типа превозни средства – коли с 4 колелета, камиони с 10 колелета и триколки/ Trikes/ с 3 колелета. Ники и Иво не са много умни и успяват да запомнят само общия брой на гумите **S**. И сега се чудят какви и колко превозни средства е имало на паркинга. Никой не може да им помогне, но вие поне, можете да намерите всички възможни комбинации от коли, камиони и триколки по общия брой на гумите – **S** , които Ники и Иво ви съобщят. **Отпечатайте в конзолата броя на вариантите.**

|  |  |
| --- | --- |
| Input | Output |
| 7 | 1 |
| 1 car and 1 trike |  |
|  |  |
| 10 | 2 |
| 1 car and 2 trikes |  |
| 1 truck |  |
| 40 | 11 |

11. Напишете скрипт, който проверява елементите на масив и връща броя на повторенията на всеки елемент в масива, във вида –

Елемент -> брой повторения

Елемент2 -> брой повторения

Елемент3 -> брой повторения

И т.н

|  |  |
| --- | --- |
| **Input** | **Output** |
| [2, 12, 2, 2, 25, 12] | 2 -> 3  12 -> 2  25 -> 1 |
| ['Welcome’, ‘to‘, ‘Welcome’, ‘to’, ‘Java’, ‘ Welcome’, ‘everyone'] | Welcome -> 3  to -> 2  Java -> 1  Everyone -> 1 |
| [1, 2, 3, 1, 4, 5, 6, 7, 6, 3, 1, 1, 4] | 1 -> 4 |
| 2 -> 1 |
| 3 -> 2 |
| 4 -> 2 |
| 5 -> 1  6 -> 2  7 -> 1 |

1. Намерете елементът, който се повтаря най-много в масив. Отпечатайте стойността му и броя на повторенията.