JS problem solving questions and answers done by B Nandiswar

Difficulty Level : Easy

1.Create a function that takes two numbers as arguments and returns their sum.

```
let a = +prompt("enter 1st number")
let b= +prompt("enter 2nd number")
function sumofab(a,b){
    return a+b;
}
console.log(sumofab(a,b))
```

o/p:

```
enter 1st number10
enter 2nd number20
30
```

2.Write a function that takes an integer minutes and converts it to seconds.

```
var min = +prompt("enter minutes  ")
function minutestoseconds(min)
{
    return min*60;
}
console.log(minutestoseconds(min))
```

o/p:

```
enter minutes  60
3600
```

3.Create a function that takes a number as an argument, increments the number by +1 and returns the result.

```
var number = +prompt("enter a number for increment  ")
function increment(number)
{
    number++;
    return number;
}
console.log(increment(number))
```

o/p:

```
enter a number for increment  4
5
```

4.Create a function that takes the age in years and returns the age in days.

```
var age = +prompt("enter age  ")
function ageyearintodays(age)
{

    return age*365;
}
console.log(ageyearintodays(age))
```

o/p:

```
enter age  10
3650
```

5.sniCreate a function that takes voltage and current and returns the calculated power.

```
var voltage = +prompt("enter voltage  ")

var current = +prompt("enter current  ")

function power(voltage,current)
{
    return voltage*current;
}

console.log(power(voltage,current));
```

o/p:

```
enter voltage 5
enter current 5
25
```

6.Write a function that returns the string "something" joined with a space " " and the given argument a.

```
var text = prompt("enter some text")
function textintogreeting(text)
{
return "something" + " " + text;
}
console.log(textintogreeting(text))
```

o/p:

```
enter some text: hi
something hi
```

7.Create a function that takes two arguments. Both arguments are integers, a and b. Return true if one of them is 10 or if their sum is 10.

```
var a=+prompt("enter a number");
var b=+prompt("enter another number");
function both_sum_same(a,b){
    return ((a===10)||(b==10)) || (a+b === 10)
}
console.log(both_sum_same(a,b));
```

```
o/p:
enter a number5
enter another number5
true
```

8.Create a function that takes two strings as arguments and returns either true or false depending on whether the total number of characters in the first string is equal to the total number of characters in the second string.

```
var str1=prompt("enter a string :")
var str2=prompt("enter a string :")

function stringsameornor(str1,str2){
    // return str1.length === str2.length ;

    var count1 = 0;
    for (let i of str1){
        count1++;
    }
    var count2=0;
    for (let j of str2){
        count2++;
    }
    return count1 == count2;
}

console.log(stringsameornor(str1,str2));
```

```
o/p:

enter a string :mutton
enter a string :prawns
true
```

9.Create a function that takes a name and returns a greeting in the form of a string. Don't use a normal function, use an arrow function.

```
var name = prompt("enter a  name : ")
var greeting = "happy Birthday ! "
let wishes = (greeting,name) => greeting+name;
console.log(wishes(greeting,name))
```

```
o/p:

enter a  name : nandi
happy Birthday ! nandi
```

10.Create a function that takes an array of 10 numbers (between 0 and 9) and returns a string of those numbers formatted as a phone number (e.g. (555) 555-5555).

```
num = [9,3,9,1,1,2,6,6,9,3]
function numberformat(num){
var a="";
var b="";
var c="";
for(var i=0 in num){
    if(i<3)
     {
     a=a+num[i];
     }
    else if (i>=3 && i<6)
    {
     b=b+num[i];
    }
    else if (i>=6 && i<=10)
    {
     c=c+num[i];
    }
}
return (a+"-"+b+"-"+c);

}
console.log(numberformat(num));
```

o/p:
939-112-6693

11.Create a function that returns an array of strings sorted by length in ascending order.
Example:
sortByLength(["a", "ccc", "dddd", "bb"]) → ["a", "bb", "ccc", "dddd"]

ans:

```
n=["a", "ccc", "dddd", "bb"]

function sortbylength(n)
{
    for(var i=0;i<n.length;i++)
{
    for(var j=0;j<n.length-i-1;j++)
    {

        if(n[j].length > n[j+1].length)
        {
            var temp=0;
            temp=n[j];
            n[j]=n[j+1]
            n[j+1]=temp;
        }
```

```
    }
}
return n;

}
console.log(sortbylength(n));
```

o/p:

```
[ 'a', 'bb', 'ccc', 'dddd' ]
```

12.Create a function that takes an array of arrays with numbers. Return a new (single) array with the largest numbers of each.
Example:
findLargestNums([[4, 2, 7, 1], [20, 70, 40, 90], [1, 2, 0]]) → [7, 90, 2]

ans:

```
var arr=[[4, 2, 7, 1], [20, 70, 40, 90], [1, 2, 0]];
var temparr = [];
for (var i of arr)
{
    var max=0;
    for(var j of i)
    {

        if(j>max)
        {
            max=j;
        }
    }
 temparr.push(max);
}
console.log(temparr);
```

o/p:
[7,90,2]

13.Create a function that takes an array of numbers and returns the second largest number.
Example:
secondLargest([10, 40, 30, 20, 50]) → 40

ans:

```
n=[10, 40, 30, 20, 50]
function secondlargest(n)
{
    for(var i=0;i< n.length;i++)
{
```

```
    for(var j = i+1;j< n.length;j++)
    {
       if(n[i] > n[j]) {
          var temp = n[j];
           n[j]=n[i];
           n[i]=temp;
       }
    }
}
var count=0;
for (var i of n)
{
    count++;
}

return "second largest number is : " + n[count-2];

}
console.log(secondlargest(n));
```

o/p:
second largest number is : 40

14.Create a function that takes an array of items, removes all duplicate items and returns a new array in the same sequential order as the old array (minus duplicates).
Example:

removeDups([1, 0, 1, 0]) → [1, 0]

removeDups(["The", "big", "cat"]) → ["The", "big", "cat"]

ans:

```
var arr = ["the","big","cat"];
let newarr = [];
for (let i = 0; i < arr.length; i++) {
    let unique = true;
    for (let j = 0; j < newarr.length; j++) {
        if (arr[i] === newarr[j]) {
            unique = false;
            break;
        }
    }
    if (unique) {
        newarr[newarr.length] = arr[i];
    }
}
console.log(newarr);
```

o/p:

["the","big","cat"]

15.Create a function that takes an array of integers as an argument and returns a unique number from that array. All numbers except unique ones have the same number of occurrences in the array.

Example:

```
findSingleNumber([2, 2, 2, 3, 4, 4, 4]) → 3
```

ans:

```
var a =[2, 2, 2, 3, 4, 4, 4];
function singlenumber(a)
{
var temparr=[]
for (var i of a)
{
    var count = 0;
    for(var j of a )
    {
        if(i==j)//here 3 == 3 will match we will get one here and all values are
above 1
        {
            count++;
        }
    }
    if(count==1)
    {
        temparr.push(i);
    }
}

return temparr;
}
console.log(singlenumber(a));
```

o/p:

```
[3]
```

16.Create a function that takes two strings as arguments and returns the number of times the first string (the single character) is found in the second string.
Example:

```
charCount("c", "Chamber of secrets") → 1
```

```
a="Chamber of secrets"
function findsinglechar(a)
{
    var count=0

    for (let i of a)
    {
        if  (i == "C")
        {
```

```
        count++;
        }
    }
   return count;
}
console.log(findsinglechar(a));
```

o/p:
1

17.Create a function that takes a string and returns the number (count) of vowels contained within it.
Example:

countVowels("Celebration") → 5

```
a="Celebration"
function countVowels(a)
{
    var count=0

    for (let i of a)
    {
      if (i == "a" || i == "e" || i=="i" ||i=="o" || i=="u")
      {
          count++;
      }
    }
   return count;
}
console.log(countVowels(a));
```

o/p:
5

18.Given a string, create a function to reverse the case. All lower-cased letters should be upper-cased, and vice versa.
Example:
reverseCase("Happy Birthday") → "hAPPY bIRTHDAY"

```
ans;
n="Happy Birthday"
function reverseCase(n){
    var a="";
for(i = 0;i<n.length;i++)
{
    if(n.charCodeAt(i) > 65 && n.charCodeAt(i) <= 90  )
    {
        a = a + n[i].toLowerCase();
    }
    else if(n[i] == " ")
    {
        a = a + " ";
```

```
    }
    else if(n.charCodeAt(i) > 91 && n.charCodeAt(i) <= 122  )
    {
        a = a + n[i].toUpperCase();
    }

}
return a;
}

console.log(reverseCase(n));
```

o/p:
hAPPY bIRTHDAY

19.Take one integer n, loop till n and pass each value to a function, create a function that takes one integer parameter, and multiply with 2 in every integer.

Input: n=5
Output: 2 4 6 8 10
Explanation: Loop start with 1 go till 5 bcoz n=5
1 x 2 =2, 2 x 2=4, 3 x 2=6 …..etc

```
var n=5;
var a=" ";

function mutliples0fn(n)
{
    for (let i =1;i<=n;i++)
    {
     a += 2*i+" ";
    }
    return a;
}
console.log(mutliples0fn(n));
```

o/p:
2 4 6 8 10

20.Create Function that will take one parameter and return type of the data.

Input: 500
Output: Integer

Input: Coding
Output: String

```
var input = 500;
var input = "Coding";
function to_identify_datatype(input)
{
    return (typeof input);
}
```

```
console.log(to_identify_datatype(input));

var input =
function to_identify_datatype(input)
{
    return (typeof input);
}
console.log(to_identify_datatype(input));
```

```
o/p:
500  number
coding string
```

21.Program to find greatest of three numbers(using ternery operator).

Input: 4 8 2
Output: 8 is gretest

ans:

```
var a=4;
var b=8;
var c=2;
var d = (a>b)&&(a>c) ? a+" is bigger": (b>c)? b+"  is bigger": c+"  is bigger";
console.log(d);
```

```
o/p:

8 is bigger
```

22 . C Program to find factorial of number.

Input: n=5
Output: 120

Explanation: 5 x 4 x 3 x 2 x 1 = 120

```
var n=+prompt("enter number for to know the factorial value : ")
function factorial(n)
{
    let sum=1;
    for(let i=1;i<=5;i++)
    {
        sum*=i;
    }
    return sum;
}
console.log(factorial(n));
```

```
o/p:

enter number for to know the factorial value : 5
120
```

23. C Program to arrange numbers in ascending order

Input: [2,3,1,5,4]
Output: [1,2,3,4,5]

Sort the Array using loop only(you can not use predefined function).

```
ans:n=[2,3,1,5,4]
function ascendingorder(n)
{
    for(var i=0;i< n.length;i++)
{
    for(var j = i+1;j< n.length;j++)
    {
       if(n[i] > n[j])
       {
          var temp = n[j];
           n[j]=n[i];
           n[i]=temp;
       }
    }
}
return "ascending order : " + n;
}


console.log(ascendingorder(n));
```

o/p:

```
[ 1, 2, 3, 4, 5 ]
```

24. Print Patter using loop.

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

ans:

```
n=+prompt("enter a range number for pattern : ")
function pattern(n)
{
    var st="";
    for(let i=1;i<=n;i++)
    {
         st = st+i+" ";
         console.log(st);
```

```
    }
}
pattern(n);
```

o/p:

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

25. C Program to Calculate the Power of a Number(using loop only).

Input: n=5, p=3
Output: 5 ^ 3 = 125
Explanation: 5 x 5 x 5 = 125

```
n=+prompt("enter a value to calculate power")
p=+prompt("enter exponential value for number")

function calculatepower(n,p)
{
    var sum=1;
    for (let i=1;i<=p;i++)
    {
        sum = sum*n;
    }
    return sum;
}
console.log(" power of number " +n +" is number "+ p+" is exponential value  : "
+ calculatepower(n,p));
```

26. Program to Check Whether a Number is Prime or Not

Input: 9
Output: 9 is not a prime no

Input: 7
Output : 7 is a prime no

ans:

```
number = +prompt("enter a number to check prime or not: ")
function PrimeorNot(number)
{
isPrime=true;
if ( number >  1 )
{
for (let i = 2; i < number; i++)
    {
```

```
        if (number % i == 0)
        {
            isPrime = false;
            break;
        }
    }
}
return isPrime?number+" is prime number":number+" is not a prime number";
}

console.log(PrimeorNot(number));
```

```
o/p:
enter a number to check prime or not: 67
67 is prime number
```

27. Program to find LCM of two numbers using while loop

```
                        Input: 15 50
                        Output: Lcm of 15 and 50 is 150.
```

```
ans:
var n1=+prompt("enter number for lcm : ")
var n2=+prompt("enter number for lcm : ")
function findlcm(n1,n2)
{
    var greater =0;
    var lcm=0;
    if (n1 > n2)
    {
        greater = n1;
    }
    else {
        greater = n2;
    }

    while (true)
    {
        if((greater%n1 == 0) && (greater%n2 == 0))
        {
            lcm = greater;
            break;
        }
        greater+=1;
    }

   return  "lcm is : " + lcm;
}
console.log(findlcm(n1,n2))

o/p:
```

```
enter number for lcm : 54
enter number for lcm : 24
216
```

28.Program to Display Characters from A to Z Using Loop with count.

Output: A1 B2 C3 D4 E5 F6 ……. Z26

ans:

```
function AtoZwithCount(){
var c=0;
for(i=65;i<=90;i++)
{
    c++;
    var a = String.fromCharCode(i);
    console.log(a+" "+ c);

}
}
console.log(AtoZwithCount());
```

o/p:
A1 B2 C3 D4 E5 F6 ……. Z26

29. Program to find a missing number

Input: n=5(length of array), arr= [5,3,1,4]
Output: 2 is missing

Using loop only(you can not use predefined function)

ans :

```
var n = 5;
var arr = [5,3,1,4];
var temp = (n * (n+1))/2;

console.log(temp);

for(var i of arr)
{
    temp = temp - i;
}
console.log(temp+ "is missing");
```

o/p:
2 is missing number

30. Program to count vowels and consonants in a given String.

Input: i am ram
Output: 3 vowels 3 consonants.

```
a="i am ram"
var vowels=0;
var consonants=0;
var gap=0;
for(var i of a)
{

    if(i == "a"||i == "e"||i == "i"||i == "o"||i == "u")
    {
      vowels++
    }
    else if(i == " "){
           gap++;
         }

    else{
        consonants++;
    }
}
console.log(vowels + " vowels " + consonants + " consonants ")
```

o/p:

3 vowels 3 consonants

31. program to insert the elements of an array for specific index.

Input: [1,2,3,4,5,7,8,9,10] , index=5
Output: [1,2,3,4,5,6,78,9,10]

ans:

```
var n=[1,2,3,4,5,7,8,9,10]
var index=5;
for(var i=0;i<n.length;i++)
{
    if(i == index)
    {
        n[i]=6;
    }

}
console.log(n);
```

o/p:

```
[
  1, 2, 3,  4, 5,
  6, 8, 9, 10
]
```

32.Reverse a number using while loop

input :123
output:321

```
a=123
var rem=0;sum=0;

while(a>0)
{
    rem=a%10;
    sum = (sum*10)+rem ;
    a=Math.floor(a/10);
}
console.log(sum)
```

o/p:
321

33.Count occurrence of number:

Input: [1,6,3,1,5,9,7,2,1,9,3,7,8,9,10] , no find=7
Output: 7 present 2 times.

```
a=[1,6,3,1,5,9,7,2,1,9,3,7,8,9,10]
find=7
var count=0;
for(let i of a)
{
    if ( i == find)
    {
        count++
    }
}
console.log(find + " present "+count+" times")
```

o/p:
7 present 2 times

Difficulty Level : medium

1.

2.Create a function that takes two numbers as arguments (num, length) and returns an array of multiples of num until the array length reaches length.
Examples :

arrayOfMultiples(7, 5) → [7, 14, 21, 28, 35]

arrayOfMultiples(12, 10) → [12, 24, 36, 48, 60, 72, 84, 96, 108, 120]

arrayOfMultiples(17, 6) → [17, 34, 51, 68, 85, 102]

ans:

```
num=+prompt("enter a number for multiple : ");
len=+prompt("enter length for multiples : ");
arr=[]
var sum=1
function multiples(num,len)
{
    for(let i=1;i<=len;i++)
        {
          sum = num * i;
          arr.push(sum);

        }
    return arr;
}
console.log(multiples(num,len));
```

o/p;
7 5
7 14 21 28 35

3.

4.

5.Create a function that moves all capital letters to the front of a word.
Examples :

capToFront("hApPy") → "APhpy"

capToFront("moveMENT") → "MENTmove"

capToFront("shOrtCAKE") → "OCAKEshrt"

ans:

```
var a = "moveMENT";
function captofront(a)
{
var str1="",str2="";
for(i of a)
{
```

```
     if((i.charCodeAt() >= 65) && (i.charCodeAt() <= 90))
    {
              str1=str1+i

    }
    else if((i.charCodeAt()>=91) && i.charCodeAt()<=122)
    {
              str2=str2+i;
    }
}
return str1+str2;
}
console.log(captofront(a))
```

o/p:

MENTmove

7.Write a function that accepts an array of strings. Return the longest string(can not use predefined function).

Input: [‘nik’, ’mikhil’, ’Cow’,’Elephant’]

Output: Elephant

ans:

```
var a=["nik", "mikhil", "Cow","Elephant"];

function longestString(a)
{
for(i of a)
{
 var max = 0; //max value updating here
  if(i.length>max)
  {
      max=i;
  }
}
return max;
}
console.log(longestString(a));
```

o/p;
Elephant

9.Write Program to remove duplicate elements in an array and sort it in descending order(can not use predefined function).

Input: [5,3,5,2,1,1,7,3,5,6]

Output: [7,6,5,3,2,1]

ans:

```
var arr = [5, 3, 5, 2, 1, 1, 7, 3, 5, 6];
let newarr = [];
for (let i = 0; i < arr.length; i++) {
    let unique = true;
    for (let j = 0; j < newarr.length; j++) {
        if (arr[i] === newarr[j]) {
            unique = false;
            break;
        }
    }
    if (unique) {
        newarr[newarr.length] = arr[i];
    }
}
console.log(newarr);

for(var i=0;i<newarr.length;i++)
{
    for(var j=i+1;j<newarr.length;j++)
    {
        if(newarr[i] < newarr[j])
        {
            var temp = newarr[j];
            newarr[j] = newarr[i];
            newarr[i] = temp;
        }
    }
}
console.log(newarr);
```

o/p:

[7,6,5,3,2,1]

10.Write a Program to Remove brackets from an algebraic expression(can not use predefined function).

Input: a + b-(9+c)=3
Output: a + b- 9+c=3

ans:

```
var a = "a + b-(9+c)=3";
function removeBrackets(a)
{
var temp = "";
for (i of a )
{
    if(i!=" ")
    {
       if((i != "(") && (i != ")"))
       {
```

```
          temp=temp+i
       }
    }
}
return temp;
}

console.log(removeBrackets(a));
```

o/p:
a+b-9+c=3

11.Write Program to remove duplicate elements in an array and sort it in Accending order(can not use predefined function).

Input: [Z, A, P, C, A, Z , K, N, C]
Output: [A, C, K,N, P, Z]

ans:

```
var arr = ["Z", "A", "P", "C", "A", "Z" , "K", "N", "C"];
let newarr = [];
for (let i = 0; i < arr.length; i++) {
    let unique = true;
    for (let j = 0; j < newarr.length; j++) {
        if (arr[i] === newarr[j]) {
            unique = false;
            break;
        }
    }
    if (unique) {
        newarr[newarr.length] = arr[i];
    }
}
console.log(newarr);
for(var i=0;i<newarr.length;i++)
{
    for(var j=i+1;j<newarr.length;j++)
    {
        if(newarr[i] > newarr[j])
        {
            var temp = newarr[j];
            newarr[j] = newarr[i];
            newarr[i] = temp;
        }
    }
}
console.log(newarr);
```

o/p:
['Z', 'A', 'P', 'C', 'K', 'N']
['A', 'C', 'K', 'N', 'P', 'Z']

13.Find sum of the Unique numbers:
Example : Let arr = [1, 2, 2, 1, 3, 5, 1];

The unique numbers are 1,2, 3, 5 so the sum should be 11.

ans;

```
var arr = [1, 2, 2, 1, 3, 5, 1];
let newarr = [];
for (let i = 0; i < arr.length; i++) {
    let unique = true;
    for (let j = 0; j < newarr.length; j++) {
        if (arr[i] === newarr[j]) {
            unique = false;
            break;
        }
    }
    if (unique) {
        newarr[newarr.length] = arr[i];
    }
}
console.log(newarr);
sum=0;
for(i of newarr)
{
    sum=sum+i;
}
console.log(sum);
```

o/p:
[1,2,3,5]
11

difficult level : Hard

1.Create a function that converts dash/underscore delimited words into camel casing. The first word within the output should be capitalized only if the original word was capitalized.

Examples :

toCamelCase("A-B-C") → "ABC"

toCamelCase("the-stealth-warrior") → "theStealthWarrior"

toCamelCase("The_Stealth_Warrior") → "TheStealthWarrior"

ans:

using string methods :

using substr() method:

```
var n = "ch_ai_ta_ny_a";

function camelcase(n){
var a=n.split("_");
var temp = "";
for (var i of a)
{
    var x = i.substr(0,1).toUpperCase() + i.substr(1);
    temp = temp+x;
}
return temp;
}
console.log(camelcase(n));
```

o/p:
ChAiTaNyA

using slice :

```
var n = "ch_ai_ta_ny_a";

function camelcase(n){
var a=n.split("_");
var temp = "";
for (var i of a)
{
    var x = i.slice(0,1).toUpperCase() + i.slice(1);
    temp = temp+x;
}
return temp;
}
console.log(camelcase(n));
```

o/p:

ChAiTaNyA

using substring() method :

```
var n = "ch_ai_ta_ny_a";

function camelcase(n){
var a=n.split("_");
var temp = "";
for (var i of a)
{
    var x = i.substring(0,1).toUpperCase() + i.substring(1);
    temp = temp+x;
}
return temp;
}
```

```
console.log(camelcase(n));
```

o/p:
Chaitanya

3.Write a function that takes a list of hours and returns the total weekly salary.
The input list hours is listed sequentially, ordered from Monday to Sunday.
A worker earns $10 an hour for the first 8 hours.
For every overtime hour, he earns $15.
On weekends, the employer pays double the usual rate, regardless how many hours were worked previously that week. For instance, 10 hours worked on a weekday would pay 80+30 = $110, but on a weekend it would pay 160+60 = $220.

Examples :
weeklySalary([8, 8, 8, 8, 8, 0, 0]) → 400
weeklySalary([10, 10, 10, 0, 8, 0, 0]) → 410
weeklySalary([0, 0, 0, 0, 0, 12, 0]) → 280

ans:

```
// const a = [8, 8, 8, 8, 8,0,0];
// a[0] = (8 * 10) + ((a[0]-8)*15);
// a[1] = (8 * 10) + ((a[1]-8)*15);
// a[2] = (8 * 10) + ((a[2]-8)*15);
// a[3] = (8 * 10) + ((a[3]-8)*15);
// a[4] = (8 * 10) + ((a[4]-8)*15);
// a[5] = (16 * 10) + ((a[5]-8)*30);
// a[6] = (16 * 10) + ((a[6]-8)*30);
// var sum = a[0]+a[1]+a[2]+a[3]+a[4]+a[5]+a[6];
// console.log(sum);

var a = [0, 0, 0, 0, 0, 12, 0];
var weekday_salary = 0;
var weekend_salary = 0;
var eight_hours = 80;
var weekday_eigthhour = 160;
var weekday_overtime = 15;
var weekend_overtime = 30
for(var i = 0;i<a.length;i++)
{
    if(a[i]>0 )
    {
        if(a[i]>=0 && a[i]<=4)
        {
            a[i] = (eight_hours + ((a[i]-8)* weekday_overtime));
            weekday_salary = weekday_salary + a[i];
        }
        else if(i >=5 && i<=6)
        {
          a[i] = (eight_hours * 2) + ((a[i]-8) * weekend_overtime);
         weekend_salary = weekend_salary + a[i];
        }
    }
```

```
}
console.log(weekday_salary+weekend_salary);
```

o/p:

280

practice and understanding :

```
// const a = [8, 8, 8, 8, 8,0,0];
// a[0] = (8 * 10) + ((a[0]-8)*15);
// a[1] = (8 * 10) + ((a[1]-8)*15);
// a[2] = (8 * 10) + ((a[2]-8)*15);
// a[3] = (8 * 10) + ((a[3]-8)*15);
// a[4] = (8 * 10) + ((a[4]-8)*15);
// a[5] = (16 * 10) + ((a[5]-8)*30);
// a[6] = (16 * 10) + ((a[6]-8)*30);
// var sum = a[0]+a[1]+a[2]+a[3]+a[4]+a[5]+a[6];
// console.log(sum);
```

6. Given a String(Note:- String Will Contain all later from A-Z except 1 letter, that letter you need to find out) :-

Input string="6 E @ f w 3 x y g N 1 o p Q A b c h i j # K l d m R T U V Z"

Output = ''S is missing from the String"

Note:- (
Time Complexity:- O(n) means only 1 loop you can use.
without using any predefined function.
)

ans:

```
var n =prompt("Enter a string: ").replaceAll(" ","").split("");

 for(i=65;i<=90;i++){
     c=0;
    for(j of n){
        if(j.charCodeAt(0)>=65 && j.charCodeAt(0)<=90 || j.charCodeAt(0)>=97 &&
j.charCodeAt(0)<=122){
            alpha=String.fromCharCode(i)
            if(alpha==j || alpha.toLowerCase()==j )
            c++;
        }
    }
    if(c==0){
    console.log(alpha +" is missing ")
    break
```

```
    }
}
```

o/p:

```
S is missing
```