1. Give a write up on Difference between copy by value and copy by reference.

| ****Copy by value**** | ****Copy by reference**** |
| --- | --- |
| While calling a function, when you pass values by copying variables, it is known as "Copy By Values." | While calling a function, in programming language instead of copying the values of variables, the address of the variables is used it is known as "Copy By References. |
| In this method, a copy of the variable is passed. | In this method, a variable itself is passed. |
| Changes made in a copy of variable never modify the value of variable outside the function. | Change in the variable also affects the value of the variable outside the function. |
| Does not allow you to make any changes in the actual variables. | Allows you to make changes in the values of variables by using function calls. |
| Values of variables are passed using a straightforward method. | Pointer variables are required to store the address of variables. |
| Original value not modified. | The original value is modified. |
| Actual and formal arguments will be created in different memory location | Actual and formal arguments will be created in the same memory location |

1. How to copy by value a composite datatype (array+objects)?

Arrays, objects, functions are all of object type which comes under composite data types. As we know variable holds data in case of composite data type it holds reference that is address of that particular value in memory.

var a= 10; // here a holds the value 10.

var b= [10,20] // b holds some address like 8023 etc.. not 10 and 20.

Therefore we can’t clone data in composite data types. To do that spread operator is used, that is three dots (…), it spreads the elements of that particular array or object and its values can be used to assign to some other variable.

Example program:

let arr1 = [1,2,4];

let arr2 = […arr1];

console.log(arr1); /// it will print [1,2,4]

arr2.push(4);

console.log(arr1); ///it will print [1,2,4]

console.log(arr2); ///it will print [1,2,4,4]