ACTIVIDADES ALGORITMOS III

1.- function a(x,y){

return 5;

}

console.log(a(5,5))

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| X | 5 |
| Y | 5 |

|  |
| --- |
| CONSOLA |
| 5 |
|  |

2.- *function* a(x,y){

z = []

z.push(x);

z.push(y);

z.push(5);

console.log(z);

return z;

}

b = a(2,2)

console.log(b);

console.log(a(6,8));

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| X | 2 |
| Y | 2 |
| Z | 2,2,5 |
| B | 2,2,5 |

|  |
| --- |
| CONSOLA |
| [2,2,5] |
| [2,2,5] |
| [6,8,5] |
| [6,8,5] |

3.- *function* a(x){

z = [];

z.push(x);

z.pop();

z.push(x);

z.push(x);

return z;

}

y = a(2);

y.push(5);

console.log(y);

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| X | 2 |
| Z | 2,2 |
| Y | 2,2,5 |

|  |
| --- |
| CONSOLA |
| [2,2,5] |

4.- *function* a(x){

if(x[0] < x[1]) {

return true;

}

else {

return false;

}

}

b = a([2,3,4,5])

console.log(b);

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| x | [2,3,4,5] |
| B | true |

|  |
| --- |
| CONSOLA |
| True |

5.- *function* a(x){

for(var i=0; i<x.length; i++){

if(x[i] > 0){

x[i] = “Coding”;

}

    }

    return x;

}

console.log(a([1,2,3,4]))

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| i | 0->1->2->3->4 |
|  |  |

|  |
| --- |
| CONSOLA |
| [coding, coding,coding, coding] |

6.- function a(x){

for(var i=0; i<x.length; i++){

if(x[i] > 5){

x[i] = “Coding”;

}

else if(x[i] < 0){

x[i] = “Dojo”;

}

    }

    return x;

}

console.log(a([5,7,-1,4]))

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| i | 0->1->2->3 |
|  |  |

|  |
| --- |
| CONSOLA |
| [5, coding, dojo, 4] |

7.- function a(x){

if(x[0] > x[1]) {

return x[1];

}

return 10;

}

b = a([5,10])

console.log(b);

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| b | 10 |
|  |  |

|  |
| --- |
| CONSOLA |
| 10 |

8.- function sum(x){

sum = 0;

for(var i=0; i<x.length; i++){

sum = sum + x[i];

console.log(sum);

}

return sum;

}

|  |  |
| --- | --- |
| VARIABLE | VALOR |
| sum | 0 |
| i | 0 |

|  |
| --- |
| CONSOLA |
| No se ejecuta la función por que no hay un llamado y las variables son locales, además no se le asignan parámetros a la función |

ALGORITMOS III PARTE 2

1.- *function* printAverage(x){

sum = 0;

// your code here

}

y = printAverage([1,2,3]);

console.log(y); // should log 2

y = printAverage([2,5,8]);

console.log(y); // should log 5

RESPUESTA

*function* printAverage(x){

sum = 0;

for (var i=0;i<x.length; i++){

sum= sum+x[i]

}

return sum/x.length

}

y = printAverage([1,2,3]);

console.log(y); // should log 2

y = printAverage([2,5,8]);

console.log(y); // should log 5

2.- *function* returnOddArray(){

// your code here

}

y = returnOddArray();

console.log(y); // should log [1,3,5,...,253,255]

Respuesta

*function returnOddArray(){*

*for(var i=1; i<=255; i+=2) {*

*console.log(i);}*

*}*

*y = returnOddArray();*

*console.log(y); // should log [1,3,5,...,253,255]*

3.- *function* squareValue(x){

// your code here

return x;

}

y = squareValue([1,2,3]);

console.log(y); // should log [1,4,9]

y = squareValue([2,5,8]);

console.log(y); // should log [4,25,64]

RESPUESTA

function squareValue(x){

for(var i=0; i<x.length; i++){

x[i]=Math.pow(x[i], 2);

}

return x;

}

y = squareValue([1,2,3]);

console.log(y); // should log [1,4,9]

y = squareValue([2,5,8]);

console.log(y); // should log [4,25,64]