Напишите функцию для оборачивания текста в тег, с которой можно было бы работать в следующем стиле:

var wrapH1=buildWrapper("H1"); // строим функцию для оборачивания текста в тег H1

var wrapP=buildWrapper("P"); // строим функцию для оборачивания текста в тег P

console.log( wrapH1("СТИХИ") );

// в консоль выводится строка "<H1>СТИХИ</H1>"

console.log( wrapP("Однажды в студёную зимнюю пору") );

// в консоль выводится строка "<P>Однажды в студёную зимнюю пору</P>"

Функция должна учитывать, что некоторые символы надо замеменять на HTML-мнемоники (а именно - символы < > ' " &):

console.log( wrapP("Вкусные M&M's") );

// в консоль выводится строка "<P>Вкусные M&amp;M&apos;s</P>"

Напишите функцию deepCopy для глубокого копирования переданного ей значения.

Функция должна получать число, строку, логическое значение, хэш или массив и возвращать его копию, включая все подхэши, подмассивы и т.д.

Напишите тесты правильной работы функции, например такие:

var h1={ a:5, b:{b1:6,b2:7}, c:[33,22], d:null, e:undefined, f:Number.NaN };

var h2=deepCopy(h1);

h1===h2 будет false

h1.a===h2.a будет true

h1.b===h2.b будет false

h1.b.b1===h2.b.b1 будет true

h1.c===h2.c будет false

h1.c[0]===h2.c[0] будет true

h1.d===h2.d будет true

h1.e===h2.e будет true

isNaN(h2.f) будет true

h2.c instanceof Array будет true

var a1=[ 5, {b1:6,b2:7}, [33,22], null, undefined, Number.NaN];

var a2=deepCopy(a1);

a1===a2 будет false

typeof(a2)===typeof(a1) будет true

a1[0]===a2[0] будет true

a1[1]===a2[1] будет false

a1[1].b1===a2[1].b1 будет true

a1[2]===a2[2] будет false

a1[2][0]===a2[2][0] будет true

a1[3]===a2[3] будет true

a1[4]===a2[4] будет true

isNaN(a2[5]) будет true

a2[2] instanceof Array будет true

var v1="sss";

var v2=deepCopy(v1);

typeof(v2)===typeof(v1) будет true

v1===v2 будет true

var z1=null;

var z2=deepCopy(z1);

typeof(z2)===typeof(z1) будет true

z1===z2 будет true

var n1=Number.NaN;

var n2=deepCopy(n1);

typeof(n2)===typeof(n1) будет true

isNaN(n2) будет true