tools used for obfuscation and deobfuscation:

OBFUSCATION:
{obfuscator.io}
variables renaming; strings extraction and encryption;
dead code injection; control flow flattening;
various code transformations; debug protection;
disable console output; domain lock.

{ByteHide}
remove indentation, spaces, newlines;
control flow fattening;
adding meaningless code;
rename variables and functions;
comments obfuscation.

DEOBFUSCATION:
{relative.im}
array map; control flow; dead code;
demangle; desequence; literal map;
member expression cleaner; rename;
simplify; string decoder.

{js-beautify}
Restores indentation, spaces,
newlines and control flow.

{obf-io.deobfuscate.io}
Unpacks arrays containing literals
and replaces all references to them;
Removes simple proxy functions,
array proxy functions and arithmetic
proxy functions;
Simplifies arithmetic expressions;
Simplifies string concatenation;
Renames unreadable hexadecimal identifiers;
Converts computed to static member
expressions and beautifies the code;
Experimental function evaluation.

OBSERVATIONS WITHOUT USE OF TOOLS FOR COMPARISON AND ANALYSIS:

COMPARISON AND ANALYSIS RESULTS USING generateRandomColor.js; isPositiveInArray.js; spacesToUnderscore.js; foundSomePeople.js; functionPerformance.js

ORIGINAL - OBFUSCATED:
{original -> obfuscator.io}
unused code insertion; renaming variable; strings extraction and encryption; number transformation; indentation, spaces, new lines removed; control flow flattening.

{original -> bytehide}
unused code insertion; renaming variables and functions; number transformation; indentation, spaces, new lines removed; control flow flattening; comments obfuscation.

OBFUSCATED - DEOBFUSCATED:
{obfuscator.io -> relative.im}
almost all unused code removed; renamed variables not restored; numbers restored; control flow, indentation, spaces, new lines restored, array map, string decoder.

{obfuscator.io -> js-beautify}
only control flow, indentation, spaces, new lines restored.

{obfuscator.io -> obf-io.deobfuscate.io}
control flow, spaces, indentation, new lines restored; unused code removed; numbers not restored; renamed variable not restored; simplifies string concatenation; array restored.

{bytehide -> relative.im}
almost all unused code removed; renamed variables not restored; number restored; control flow, indentation, spaces, new lines restored, array map, string decoder.

{bytehide -> js-beautify}
only control flow, indentation, spaces, new lines restored.

{bytehide -> obf-io.deobfuscate.io}
control flow, spaces, indentation, new lines restored; unused code removed; numbers not restored; renamed variable not restored; simplifies string concatenation; array restored.

ORIGINAL - DEOBFUSCATED:
{original --> [obfuscator.io -> relative.im]}
the deobfuscated result is very similar to the original one, traces of unused code and not original variables names remain.

{original --> [obfuscator.io -> js-beautify]}
in the deobfuscated code the only difference with the obfuscated one is the indentation, spaces, new lines and control flow restored.

{original --> [obfuscator.io -> obf-io.deobfuscate.io]}
the deobfuscated result is very similar to the original one, renamed variables and hexadecimal numbers not restored.

{original --> [bytehide -> relative.im]}
the deobfuscated result is very similar to the original one, traces of unused code and not original variables names remain.

{original --> [bytehide -> js-beautify]}
in the deobfuscated code the only difference with the obfuscated one is the indentation, spaces, new lines and control flow restored.

{original --> [bytehide -> obf-io.deobfuscate.io]}
the deobfuscated result is very similar to the original one, renamed variables and hexadecimal numbers not restored.

OBSERVATIONS AFTER USE OF BASH COMMAND DIFF FOR COMPARISON AND ANALYSIS:

COMPARISON AND ANALYSIS RESULTS USING concatEvenWords.js; putSpace.js; max_min.js; generateUUID.js; randomBoolean.js

ORIGINAL - OBFUSCATED:
{original -> obfuscator.io}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

{original -> bytehide}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

Observations:
They are different because the original code is transformed a lot through some obfuscation methodologies and techniques. The two codes appear different but the result obtained from the execution of the two functions is identical.

OBFUSCATED - DEOBFUSCATED:
{obfuscator.io -> relative.im}

- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean

diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

{obfuscator.io -> js-beautify}

- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean

diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

{obfuscator.io -> obf-io.deobfuscate.io}

- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean

diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

{bytehide -> relative.im}

- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean

diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

{bytehide -> js-beautify}

- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean

diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

{bytehide -> obf-io.deobfuscate.io}

- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean

diff default command
perceives the two files as different so it reports the all code.
diff -q, --brief report only when files differ
return that the two files are different.

Observations:

They are different because the obfuscated code is transformed a lot through some deobfuscation methodologies and techniques trying to bring it back to its original shape. The two codes appear different but the result obtained from the execution of the two functions is identical.

ORIGINAL - DEOBFUSCATED:
{original --> [obfuscator.io -> relative.im]}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
perceives that the code is moved because there is extra code and also shows that some words are different from the original code because they have not been restored.
diff -q, --brief report only when files differ
return that the two files are different because of extra code and not restored words to the original form.

{original --> [obfuscator.io -> js-beautify]}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
restoring only the original code structure by adding spaces, new lines and indentations, the two codes are perceived as different.
diff -q, --brief report only when files differ
return that the two files are different because it restores only the original code structure by adding spaces, new lines and indentations.

{original --> [obfuscator.io -> obf-io.deobfuscate.io]}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
perceives the two codes as similar but with some differences due to the fact that the variables and numbers have not been restored, furthermore some parts of the code are positioned differently or presented in a more concise way.
diff -q, --brief report only when files differ
return that the two files are different because of variables and numbers not restored, different position of some parts of the code and for the fact that sometimes some parts of code are reported in a more concise way.

{original --> [bytehide -> relative.im]}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
perceives that the code is moved because there is extra code and also shows that some words are different from the original code because they have not been restored and array structure is slightly different.
diff -q, --brief report only when files differ
return that the two files are different because of extra code, not restored words to the original form and array structure is slightly different.

{original --> [bytehide -> js-beautify]}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
restoring only the original code structure by adding spaces, new lines and indentations, the two codes are perceived as different.
diff -q, --brief report only when files differ
return that the two files are different because it restores only the original code structure by adding spaces, new lines and indentations.

{original --> [bytehide -> obf-io.deobfuscate.io]}
- concatEvenWords; putSpace; max_min; generateUUID; randomBoolean
diff default command
perceives the two codes as similar but with some differences due to the fact that the variables and numbers have not been restored, furthermore some parts of the code are positioned differently or presented in a more concise way.
diff -q, --brief report only when files differ
return that the two files are different because of variables and numbers not restored, different position of some parts of the code and for the fact that sometimes some parts of code are reported in a more concise way.

OBSERVATIONS AFTER USE OF BASH COMMAND SDIFF FOR COMPARISON AND ANALYSIS:

COMPARISON AND ANALYSIS RESULTS USING randomLetter.js; getDuplicates.js; include.js; countVowel.js; randomNumberInRange.js

ORIGINAL - OBFUSCATED:
{original -> obfuscator.io}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

{original -> bytehide}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

Observations:
They are different because the original code is transformed a lot through some obfuscation methodologies and techniques. The two codes appear different but the result obtained from the execution of the two functions is identical.

OBFUSCATED - DEOBFUSCATED:
{obfuscator.io -> relative.im}

- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

{obfuscator.io -> js-beautify}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

{obfuscator.io -> obf-io.deobfuscate.io}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

{bytehide -> relative.im}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

{bytehide -> js-beautify}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

{bytehide -> obf-io.deobfuscate.io}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two files as different.

Observations:
They are different because the obfuscated code is transformed a lot through some deobfuscation methodologies and techniques trying to bring it back to its original shape. The two codes appear different but the result obtained from the execution of the two functions is identical.

ORIGINAL - DEOBFUSCATED:
{original --> [obfuscator.io -> relative.im]}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives that the code is moved because there is extra code and also shows that some words are different from the original code because they have not been restored and array structure is slightly different.

{original --> [obfuscator.io -> js-beautify]}

- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the similarity of some parts of the two codes but considers them mostly different because only the structure of the original code has been restored with spaces, new lines and indentations.

{original --> [obfuscator.io -> obf-io.deobfuscate.io]}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two codes as similar but with some differences due to the fact that the variables and numbers have not been restored, furthermore some parts of the code are positioned differently or presented in a more concise way.

{original --> [bytehide -> relative.im]}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives that the code is moved because there is extra code and also shows that some words are different from the original code because they have not been restored and array structure is slightly different.

{original --> [bytehide -> js-beautify]}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the similarity of some parts of the two codes but considers them mostly different because only the structure of the original code has been restored with spaces, new lines and indentations.

{original --> [bytehide -> obf-io.deobfuscate.io]}
- randomLetter; getDuplicates; include; countVowel; randomNumberInRange
sdiff default command
perceives the two codes as similar but with some differences due to the fact that the variables and numbers have not been restored, furthermore some parts of the code are positioned differently or presented in a more concise way.

OBSERVATIONS AFTER USE OF TOOL KOMPARE FOR COMPARISON AND ANALYSIS:

COMPARISON AND ANALYSIS RESULTS USING calculateAge.js; factorial.js; generateStrongPassword.js; round.js; mergeTwoObjectsIntoOne.js

ORIGINAL - OBFUSCATED:
{original -> obfuscator.io}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different.

{original -> bytehide}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, or in green and red, which green identifies a removal, therefore perceives the two files as different.

OBFUSCATED - DEOBFUSCATED:
{obfuscator.io -> relative.im}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different.

{obfuscator.io -> js-beautify}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different.

{obfuscator.io -> obf-io.deobfuscate.io}
- calculateAge; factorial (giving problems during deobfuscation, it cannot be compared and analyzed); generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different.

{bytehide -> relative.im}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different.

{bytehide -> js-beautify}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, or in green and red, which green identifies a removal, therefore perceives the two files as different.

{bytehide -> obf-io.deobfuscate.io}
- calculateAge; factorial (giving problems during deobfuscation, it cannot be compared and analyzed); generateStrongPassword; round; mergeTwoObjectsIntoOne

highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different.

ORIGINAL - DEOBFUSCATED:
{original --> [obfuscator.io -> relative.im]}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne
highlights the two files entirely in red, the color that identifies a change, it is changed because of extra code, not restored words to the original form and array structure is slightly different, some parts are also highlighted in purple to identify code additions.

{original --> [obfuscator.io -> js-beautify]}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne
highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different, some parts are also highlighted in purple to identify code additions.

{original --> [obfuscator.io -> obf-io.deobfuscate.io]}
- calculateAge; factorial (giving problems during deobfuscation, it cannot be compared and analyzed); generateStrongPassword; round; mergeTwoObjectsIntoOne
highlights the two files entirely in red, the color that identifies a change, it is changed due to the fact that the variables and numbers have not been restored, furthermore some parts of the code are positioned differently or presented in a more concise way. Sometimes it is also highlighted that there are parts of code that are unchanged.

{original --> [bytehide -> relative.im]}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne
highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different, some parts are also highlighted in purple to identify code additions.

{original --> [bytehide -> js-beautify]}
- calculateAge; factorial; generateStrongPassword; round; mergeTwoObjectsIntoOne
highlights the two files entirely in red, the color that identifies a change, therefore perceives the two files as different, some parts are also highlighted in purple to identify code addition.

{original --> [bytehide -> obf-io.deobfuscate.io]}
- calculateAge; factorial (giving problems during deobfuscation, it cannot be compared and analyzed); generateStrongPassword; round; mergeTwoObjectsIntoOne
highlights the two files entirely in red, the color that identifies a change, it is changed due to the fact that the variables and numbers have not been restored, furthermore some parts of the code are positioned differently or presented in a more concise way. Sometimes it is also highlighted that there are parts of code that are unchanged.