# 留学生2年　JavaScript　卒業単位認定試験

## 次の問題を読み、正しいものには○を、誤りには×を記入しなさい。 （20点　@2点）

1. 変数の宣言は、Arrayを使う （　×　）
2. 文字列は、[　と　]　を使って囲む。 （　×　）
3. 変数aaaを宣言（　var　　aaa;　）したとき、aaaの初期値は、０である。 （　×　）
4. 変数の初期化を行わない時、変数の値は0となる。 （　×　）
5. 繰り返しを行う時には、if文か、switch文を使う。 （　×　）
6. 繰り返しでbreakを使用すると、以降の処理を実行せず先頭に戻る。 （　×　）
7. 条件判断をした結果（true／false）に応じて、処理を二つに分けるには、switch制御命令を使用する。 （　×　）
8. 繰り返しには、if、for、while、switchなどのキーワードが使用される。 （　×　）
9. break命令は、ループと組み合わせて使われ、この命令以降の命令を実行せずに繰り返しを続ける制御命令である。 （　×　）
10. 制御命令（if、for、while、switch）の条件式の後には必ず「；」（セミコロン）を書く （　×　）

## 次のプログラムを読み、ループの処理（A）が何回実行されるか答えなさい。（20点　@5点）

|  |  |  |
| --- | --- | --- |
| 問題 | プログラム | 解答 |
| 1 | function foo( j){  for ( var x = 1 ; x<10 ; x++ ){  document.write( "hello <br>");　　　　←（A)  }  }  foo( 3); | 9回 |
| 2 | function ifs( param ){  if ( param > 0 )  if ( param > 5 )  ans = true ;  else  ans = fase ;  ans = true ;  return ans ;  }  RET = ifs ( 15 ); |  |
| 3 | function Bob( No) {  var end = No % 10;  while ( end > 0 ){  document.write("hello<BR>");　　　　　　　←（A)  end--;  }  }  Bob( 学績番号); | 学績番号の末尾の1桁の回数 |
| 4 | function Green(param) {  var max;  switch (param) {  case 1:  case 2:  case 3:  max = 2;  case 4:  case 5:  max = 3;  default:  max = 4;  }  for( var i = 0 ; i<=max; i++){  document.write( "hello world<br>");　　　　←（A)  }  }  Green( 学績番号 % 5 ); | 5回 |

## 次のプログラムを実行した時、画面上に表示される値を答えなさい。

## 次の関数の結果（変数RET）がどのようになるか答えなさい。問題文中の【学籍番号】には自分の学籍番号が入るものとして答えなさい。 （20点　@4点）

|  |  |  |
| --- | --- | --- |
| 問題 | プログラム | 解答 |
| 1 | function doAction( p1, p2 ){  var ans;  ans = p1 - p2 ;  return ans;  }  RET = doAction( 100, 2); | 98 |
| 2 | function doAction() {  var ary = new Array( 5);  for ( var j = 0; j < 5 ; j++ ){  ary[j] = 2:  }  for ( var k = 1 ; k < 5 ; k++){  ary[k] = ary[k] + ary[k-1];  }  var ans ;  ans = ary[3] + ary[4];  return ans;  }  RET = doAction(); | 18 |
| 3 | function binPower( n){  var ans = 1;  for( var i = 1; i <= n ; i++){  ans \*= 2;  }  return ans;  }  RET = binPower( 10); | 1024 |

## 画面上に、「hellow word」の文字を10回表示するJavaScriptを書いた。この時の誤りを指摘し、どのように修正をすればよいか答えなさい。

1. 次のforの命令について不具合を修正しなさい。

【ソースコード】



【HTMLプレビュー】



【解答欄】

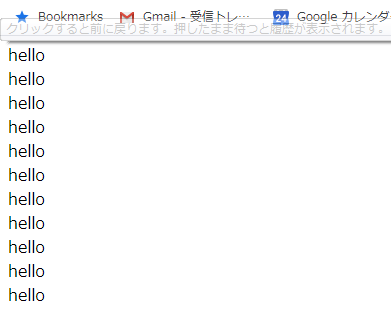
|  |  |  |
| --- | --- | --- |
| 行 | 誤りの部分 | 正しい命令 |
|  |  |  |

1. 次のwhile命令について不具合を修正しなさい。

【ソースコード】



【HTMLプレビュー】



【解答欄】

|  |  |  |
| --- | --- | --- |
| 行 | 誤りの部分 | 正しい命令 |
|  |  |  |

## 次の関数の仕様を読み、関数を作成しなさい。 （40点　@4点）

|  |  |  |
| --- | --- | --- |
| 問題 | 仕様 | |
| 1 | 関数名 | foo |
| パラメータ | p |
| 処理 | もし、パラメータｐが１のとき、変数ansに「おはよう」をセットする  　　　パラメータｐが１ではない時、変数ansに「こんにちわ」をセットする |
| 戻り値 | ans |
| 解答欄 | | |
| function foo( p){  var ans;  if ( p == 1 ) {  ans = "おはよう";  }else {  ans = "こんにちわ";  }  return ans;  } | | |
| 問題 | 仕様 | |
| 2 | 関数名 | kaijo |
| パラメータ | n |
| 処理 | 変数ansを宣言する  もし､n>1なら　　ans　←　n\*kaijo(n-1)  それ以外なら、 ans　←　1 |
| 戻り値 | ans |
| 解答欄 | | |
| function kaijo( n){  　　var ans;  　　if ( n > 1 ) {  　　　　ans = n \* kaijo( n -1 );  　　} else {  　　　　ans = 1;  　　}  　　return ans;  } | | |
| 問題 | 仕様 | |
| 3 | 関数名 | showSectionForName |
| パラメータ | sectionIdName |
| 処理 | var item = document.getElementById( sectionIdName);  item.style.display = "block"; |
| 戻り値 | なし |
| 解答欄 | | |
| function showSectionfForName( sectionIdName ){  　　var item = document.getElementById( sectionIdName);  　　item.style.display = “block”;  } | | |