

Catedráticos: Ing. Bayron López, Ing. Erick Navarro y Ing. Edgar Saban

Tutores académicos: Mike Gutiérrez, Javier Navarro, Julio Arango.

Log de Cambios CAAS

Segundo proyecto de laboratorio

1 Método equals: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
28	sección 5.1.2.2.1 párrafo 1	El método equals() define una noción de igualdad de objeto, que se basa en el valor, no en la comparación, en la comparación.	El método equals(Object o) define una noción de igualdad de objeto, que se basa en la referencia, no en la comparación, en la comparación. Definirá la igualdad dada por el objeto al que pertenece y el objeto o recibido como parámetro.

2 Anulación de función read: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
75	sección 7.6.10.4 párrafo 1	Lectura de datos de parte del usuario	Esta función no deberá ser implementada.
68	Ejemplo 23, línea 4	int numero = read_int();	int numero = 5;

3 Case sensitive : 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
114	sección 15.2 inciso 7	Todos los lenguajes implementados no hacen distinción entre mayúsculas y minúsculas, es decir no son case sensitive.	El lenguaje Coline es case sensitive.

4 Longitud de String: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
27	tabla 2, fila 6	[-2147483648, 2.147.483.647] caracteres	[0, 2.147.483.647] caracteres

5 Palabras reservadas Coline: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
27	tabla 3	graph_dot	graph
27	tabla 3	read_console	no debería de estar
27	tabla 3	message	no debería de estar
27	tabla 3	Object	no debería de estar

6 Expresión regular de decimales: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
32	sección 6.8	Decimales: [0-9]+("." [WHITESPACE 0-9]+)?	Decimales: [0-9]+("." [0-9]+)?
96	sección 8.1.9	Decimales: [0-9]+("." [WHITESPACE 0-9]+)?	Decimales: [0-9]+("." [0-9]+)?

7 Sintaxis de métodos: 25 de marzo de 2019 10:50 horas:

Pag .	Localidad	Donde dice	Debería de decir
45	Sintaxis 6	<pre> <method_declaration>::=<method_header> <statement_block> <method_header> ";" <method_header>::= <method_modifiers> <result> <method_declarator> <method_declarator>::= <method_declarator> "[" "]" id "(" <formal_parameters> ")"</pre>	<pre> <method_declaration>::=<method_header> <statement_block> <method_header> ";" <method_header>::= <method_modifiers> <result> <method_declarator> id "(" <formal_parameters> ")" <method_declarator>::= <method_declarator> "[" "]" "[" "]" ε</pre>

8 Firma de métodos: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
45	sección 7.2.5 Inciso 3 de consideraciones	Es un error de tiempo de compilación para el cuerpo de una clase el declarar como miembros dos métodos con el mismo nombre	Es un error de tiempo de compilación para el cuerpo de una clase <u>el declarar como miembros dos métodos con la misma firma</u>

9 Palabra reservada @override: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
49	sección 7.2.5.5.1	Un método de instancia m_1 , declarado en la clase C , anula otro método de instancia m_2 , declarado en la clase A si todos los siguientes son verdaderos:	Un método de instancia m_1 , declarado en la clase C , anula otro método de instancia m_2 , <u>anteponiendo al método @Override, permitiendo así, que se tengan dos métodos con la misma firma, el cual estará declarado en la clase</u> A si todos los siguientes son verdaderos:

10 Operador de módulo C3D: 25 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
97	sección 8.2.2 tabla 15		% modulo (es decir, falto definir la operación de modulo que realiza el modulo tal como se definió en Coline)

11 Precedencia xor: 27 de marzo de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
112	Apéndice A tabla 36		Falta el nivel: <u>4.5 ^ xor izq</u>

12 Sentencia Leer archivo: 1 de abril de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
74	sección 7.6.10.2	Esta función será encargada de leer un archivo y retornará el contenido del mismo.	Esta función será encargada de leer un archivo <u>del lado del servidor</u> . <u>Para su traducción a código 3D se</u> <u>utilizará la misma palabra reservada</u> <u>enviando el puntero de la cadena como</u> <u>parámetro que representa la ruta</u> <u>relativa al archivo de aplicación</u> <u>(server.js), y el intérprete 3D</u> <u>realizará el acceso al sistema de</u> <u>archivos en el servidor y retornará</u> <u>el contenido del mismo.</u>

13 Sentencia write_file: 1 de abril de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
74	sección 7.6.10.3	Esta función será encargada de escribir un archivo recibiendo la ruta y el contenido del mismo.	Esta función será encargada de escribir un archivo recibiendo la ruta y el contenido del mismo <u>del lado del servidor</u> . Para su traducción a código 3D se utilizará la misma palabra reservada enviando el puntero de la cadena que representa la ruta <u>relativa al archivo de aplicación (server.js)</u> , así como el que representa el contenido como parámetros y el intérprete 3D realizará el acceso al sistema de archivos en <u>el servidor</u> .

14 Sentencia graph: 1 de abril de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
75	sección 7.6.10.5	La sentencia recibirá la ruta de la imagen .png a generar y el contenido .dot que representa la imagen.	La sentencia recibirá la ruta <u>del lado del servidor</u> de la imagen .png a generar y el contenido .dot que representa la imagen. Para su traducción a código 3D se utilizará la misma palabra reservada enviando el puntero de la cadena que representa la ruta <u>así como el que representa el contenido dot como parámetros y el intérprete 3D realizará el acceso al sistema de archivos en el servidor</u> .

15 This: 1 de abril de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
76	sección 7.7.1 Consideraciones	Cuando se usa como una expresión, la palabra clave <code>this</code> denota un valor que es una referencia al objeto para el que se invocó el método de instancia o al objeto que se está construyendo.	Cuando se usa como una expresión, la palabra clave <code>this</code> denota un valor que es una referencia al objeto para el que se invocó el método de instancia o al objeto que se está construyendo, <u>lo que permite acceder a los miembros para obtener su valor o incluso modificarlo.</u>

16 Acceso a miembros: 1 de abril de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
76	sección 7.7.5	Se define la forma en la que se debe acceder a campos de un objeto o arreglo.	Se define la forma en la que se debe acceder a campos de un objeto o arreglo <u>para obtener el valor o bien modificarlo.</u>

17 Palabras reservadas: 1 de abril de 2019 10:50 horas:

Pag.	Localidad	Donde dice	Debería de decir
96	tabla 13		<code>null</code> (representa a la nada)

18 Sentencia `graph`: 2 de abril de 2019 14:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
75	sección 7.6.10.5 Sintaxis 35	<code>"graph" "("</code> <code>string_literal ","</code> <code>string_literal ")"</code>	<code>"graph" "("</code> <code><expression></code> <code>","</code> <code><expression></code> <code>)"</code>

19 Sentencia ifFalse: 2 de abril de 2019 23:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
99	sección 8.2.8.2	Sentencia que indicará un salto a la etiqueta que se especifique si se cumple la condición.	Sentencia que indicará un salto a la etiqueta que se especifique si <u>no</u> se cumple la condición.

20 Declaración en código intermedio: 4 de abril de 2019 8:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
97	sección 8.2.1	"var" id ";"	"var" <id_list> ";" "var" "stack" "[" "]" ";" "var" "heap" "[" "]" ";"

21 Sintaxis write_file y read_file: 4 de abril de 2019 8:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
74	sección 7.6.10.2 Sintaxis 32	"read_file" "(" String_literal ")"	"read_file" "("<expression> ")"
74	sección 7.6.10.3 Sintaxis 33	"write_file" "(" string_literal "," string_literal ")"	"write_file" "(" <expression> "," <expression> ")"

22 Sintaxis de inicialización de arreglos: 5 de abril de 2019 15:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
54	sección 7.3.4	Al momento de declarar arreglos será necesario definir el tipo de dato de los elementos que contendrá.	Al momento de declarar arreglos será necesario definir el tipo de dato de los elementos que contendrá. Y la inicialización se podrá de hacer de dos formas, como se observa en el Ejemplo 12.

23 Sintaxis declaración de arreglos: 8 de abril de 2019 21:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
53	sección 7.3.3 Ejemplo 11 línea 3	<code>int[] ia = new int[101];</code>	<code>int ia[] = new int[101];</code>
54	sección 7.3.5 Ejemplo 13 línea 5 y 6	<code>ColoredPoint[] cpa = new ColoredPoint[10]; Point[] pa = cpa;</code>	<code>ColoredPoint cpa [] = new ColoredPoint[10]; Point pa [] = cpa;</code>
55	sección 7.3.6 Ejemplo 14 línea 3	<code>int[][] arr = new int[2][3];</code>	<code>int arr [][]= new int[2][3];</code>
76	sección 7.7.1 Ejemplo 36 línea 2	<code>int[] v;</code>	<code>int v [];</code>

24 Sintaxis declaración de métodos 3D: 8 de abril de 2019 21:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
100	sección 8.2.9 Sintaxis 57	<code>"void" id "(" ")"</code> <code><3D_statement_block></code>	<code>"proc" id "begin"</code> <code><3D_statement_block> "end"</code>
101	sección 8.2.9 Ejemplo 65	<pre>void metodo1() { L2: t1 = t1 + 1; t2 = t1 + 0; t3 = 10 + 5; ifFalse (t2 < t3) goto L2; t5 = t5 + t4; t6 = t5; L2: }</pre>	<pre>proc metodo begin L2: t1 = t1 + 1; t2 = t1 + 0; t3 = 10 + 5; ifFalse (t2 < t3) goto L2; t5 = t5 + t4; t6 = t5; L2: end</pre>

25 Sintaxis invocación de métodos 3D: 8 de abril de 2019 21:00 horas:

Pag.	Localidad	Donde dice	Debería de decir
100	sección 8.2.9 Sintaxis 57	<code>"call" id "(" ")" ";"</code>	<code>"call" id ";"</code>
101	sección 8.2.9 Ejemplo 65	<code>call metodo1();</code>	<code>call metodo1;</code>