

Anexo

Prueba 1

```
let number n1;
let boolean l1;
let string cad;
let number n2;
let boolean l2;

alert ("PdL");
input (esto_es_un_nombre_de_variable_global_de_tipo_entero);
input (n1);
l1 = l2;
if (l1&& l2) cad = "hello";
n2 = n1 - 378;

alert(      33
          -
          n1
          -
          n2);
function boolean ff(boolean ss)
{
    l2 = l1;
    if (l2) l1 = ff (ss);
    varglobal = 8888;
    return (ss);
}
if (ff(l1)) alert (varglobal);
```

Parse y Arbol Sintactico

Parse

```
Ascendente 27 24 28 24 29 24 27 24 28 24 44 39 36 33 31 19 25 20 25 20 25 40 39 36 33
31 17 25 40 39 36 33 40 39 36 32 31 44 39 36 33 31 17 23 40 39 43 38 36 33 31 17 25 43
39 40 38 40 38 36 33 31 19 25 28 9 6 28 14 11 7 40 39 36 33 31 17 25 40 39 36 33 31 40
39 36 33 31 48 45 42 39 36 33 31 17 23 43 39 36 33 31 17 25 40 39 36 33 31 41 39 36 33
31 49 21 25 16 15 15 15 15 8 5 40 39 36 33 31 48 45 42 39 36 33 31 40 39 36 33 31 19 23
4 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1
```

Arbol Sintactico

Árbol resultado de:

Gramática: D:\OneDrive - Universidad Politécnica de Madrid\Universidad\3º\Procesadores de Lenguajes\Práctica\ProcesadoresLenguajes\Pruebas\ASintactico-Tests\Herramienta Vast\aSintax\GramaticaVast.txt

Parse: D:\OneDrive - Universidad Politécnica de Madrid\Universidad\3º\Procesadores de Lenguajes\Práctica\ProcesadoresLenguajes\docs\Analizador Sintáctico\Pruebas realizadas\Prueba1\Parse.txt

```
Z (1)
P (2)
B (24)
let
T (27)
    number
    identificador
    puntoYcoma
P (2)
B (24)
let
T (28)
    boolean
    identificador
    puntoYcoma
P (2)
B (24)
let
T (29)
    string
    identificador
    puntoYcoma
P (2)
B (24)
let
T (27)
    number
    identificador
    puntoYcoma
P (2)
B (24)
let
T (28)
    boolean
    identificador
    puntoYcoma
P (2)
B (25)
S (19)
    alert
    abrirParentesis
E (31)
R (33)
U (36)
V (39)
W (44)
```

cadena
cerrarParentesis
puntoYcoma

P (2)
B (25)
S (20)
 input
 abrirParentesis
 identificador
 cerrarParentesis
 puntoYcoma

P (2)
B (25)
S (20)
 input
 abrirParentesis
 identificador
 cerrarParentesis
 puntoYcoma

P (2)
B (25)
S (17)
 identificador
 asignacion
 E (31)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador

 puntoYcoma

P (2)
B (23)
 if
 abrirParentesis
 E (31)
 R (32)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador

 opLogico1
 U (36)
 V (39)
 W (40)
 identificador

 cerrarParentesis
 S (17)
 identificador
 asignacion
 E (31)

R (33)
U (36)
V (39)
W (44)
cadena

puntoYcoma

P (2)
B (25)
S (17)
identificador
asignacion
E (31)
R (33)
U (36)
V (38)
V (39)
W (40)
identificador
opAritmetico2
W (43)
cteEntera

puntoYcoma

P (2)
B (25)
S (19)
alert
abrirParentesis
E (31)
R (33)
U (36)
V (38)
V (38)
V (39)
W (43)
cteEntera
opAritmetico2
W (40)
identificador
opAritmetico2
W (40)
identificador

cerrarParentesis
puntoYcoma

P (3)
F (5)
I (6)
function
H (9)
T (28)
boolean
identificador

J (7)
 abrirParentesis
 A (11)
 T (28)
 boolean
 identificador
 K (14)
 lambda
 cerrarParentesis
 G (8)
 abrirCorchete
 C (15)
 B (25)
 S (17)
 identificador
 asignacion
 E (31)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador
 puntoYcoma
 C (15)
 B (23)
 if
 abrirParentesis
 E (31)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador
 cerrarParentesis
 S (17)
 identificador
 asignacion
 E (31)
 R (33)
 U (36)
 V (39)
 W (42)
 identificador
 abrirParentesis
 L (45)
 E (31)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador
 Q (48)

lambda

cerrarParentesis

puntoYcoma

C (15)

B (25)

S (17)

identificador

asignacion

E (31)

R (33)

U (36)

V (39)

W (43)

cteEntera

puntoYcoma

C (15)

B (25)

S (21)

return

X (49)

E (31)

R (33)

U (36)

V (39)

W (41)

abrirParentesis

E (31)

R (33)

U (36)

V (39)

W (40)

identificador

cerrarParentesis

puntoYcoma

C (16)

lambda

cerrarCorchete

P (2)

B (23)

if

abrirParentesis

E (31)

R (33)

U (36)

V (39)

W (42)

identificador

abrirParentesis

L (45)

E (31)

R (33)
U (36)
V (39)
W (40)
identificador

Q (48)
lambda
cerrarParentesis

cerrarParentesis
S (19)
alert
abrirParentesis
E (31)
R (33)
U (36)
V (39)
W (40)
identificador

cerrarParentesis
puntoYcoma
P (4)
lambda

Prueba 2

```
let string texto;
function print (string msg)
{
    alert (msg);
}
function pideTexto ()
{
    alert ("Introduce un texto");
    input (texto);
}
pideTexto();
let string textoAux;
textoAux = texto;
print (textoAux);
```

Parse y Arbol Sintactico

Parse

Ascendente 29 24 10 6 29 14 11 7 40 39 36 33 31 19 25 16 15 8 5 10 6 12 7 44 39 36 33
31 19 25 20 25 16 15 15 8 5 46 18 25 29 24 40 39 36 33 31 17 25 40 39 36 33 31 48 45 18
25 4 2 2 2 3 3 2 1

Arbol Sintactico

Árbol resultado de:

Gramática: D:\OneDrive - Universidad Politécnica de Madrid\Universidad\3º\Procesadores de Lenguajes\Práctica\ProcesadoresLenguajes\Pruebas\ASintactico-Tests\Herramienta Vast\asintax\GramaticaVast.txt

Parse: D:\OneDrive - Universidad Politécnica de Madrid\Universidad\3º\Procesadores de Lenguajes\Práctica\ProcesadoresLenguajes\docs\Analizador Sintáctico\Pruebas realizadas\Prueba2\Parse.txt

```
Z (1)
P (2)
B (24)
let
T (29)
    string
identificador
puntoYcoma
P (3)
F (5)
I (6)
function
H (10)
    lambda
identificador
J (7)
abrirParentesis
A (11)
T (29)
    string
identificador
K (14)
    lambda
cerrarParentesis
G (8)
abrirCorchete
C (15)
B (25)
S (19)
    alert
    abrirParentesis
E (31)
    R (33)
        U (36)
            V (39)
                W (40)
                    identificador
                cerrarParentesis
                puntoYcoma
C (16)
    lambda
cerrarCorchete
P (3)
F (5)
I (6)
```

```
function
H (10)
    lambda
    identificador
J (7)
    abrirParentesis
    A (12)
        lambda
        cerrarParentesis
G (8)
    abrirCorchete
    C (15)
    B (25)
        S (19)
            alert
            abrirParentesis
            E (31)
            R (33)
            U (36)
            V (39)
            W (44)
            cadena
            cerrarParentesis
            puntoYcoma
C (15)
    B (25)
    S (20)
        input
        abrirParentesis
        identificador
        cerrarParentesis
        puntoYcoma
    C (16)
        lambda
        cerrarCorchete
P (2)
    B (25)
    S (18)
        identificador
        abrirParentesis
        L (46)
            lambda
            cerrarParentesis
            puntoYcoma
    P (2)
        B (24)
            let
            T (29)
                string
            identificador
            puntoYcoma
    P (2)
        B (25)
```

S (17)
 identificador
 asignacion
 E (31)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador

 puntoYcoma
P (2)
 B (25)
 S (18)
 identificador
 abrirParentesis
 L (45)
 E (31)
 R (33)
 U (36)
 V (39)
 W (40)
 identificador

 Q (48)
 lambda
 cerrarParentesis
 puntoYcoma
P (4)
 lambda

Prueba 3

```
let number x;
let number z;
let boolean b;
input (x);
alert (x);
input (z);
alert (x+z);
b=x!=z;if (b)
x =
  x + 6
  + z
  - 1
  - (2
  - y
  - 6);
```

Parse y Arbol Sintactico

Parse

Ascendente 27 24 27 24 28 24 20 25 40 39 36 33 31 19 25 20 25 40 39 40 37 36 33 31 19
25 40 39 36 40 39 35 33 31 17 25 40 39 36 33 31 40 39 43 37 40 37 43 38 43 39 40 38 43
38 36 33 31 41 38 36 33 31 17 23 4 2 2 2 2 2 2 2 2 2 1

Arbol Sintactico

Árbol resultado de:

Gramática: D:\OneDrive - Universidad Politécnica de Madrid\Universidad\3º\Procesadores de Lenguajes\Practica\ProcesadoresLenguajes\Pruebas\ASintactico-Tests\Herramienta Vast\aSintax\GramaticaVast.txt

Parse: D:\OneDrive - Universidad Politécnica de Madrid\Universidad\3º\Procesadores de Lenguajes\Practica\ProcesadoresLenguajes\docs\Analizador Sintactico\Pruebas realizadas\Prueba3\Parse.txt

```
Z (1)
P (2)
B (24)
let
T (27)
    number
    identificador
    puntoYcoma
P (2)
B (24)
let
T (27)
    number
    identificador
    puntoYcoma
P (2)
B (24)
let
T (28)
    boolean
    identificador
    puntoYcoma
P (2)
B (25)
S (20)
    input
    abrirParentesis
    identificador
    cerrarParentesis
    puntoYcoma
P (2)
B (25)
S (19)
    alert
    abrirParentesis
E (31)
R (33)
U (36)
V (39)
W (40)
    identificador
    cerrarParentesis
    puntoYcoma
P (2)
B (25)
S (20)
```

input
abrirParentesis
identificador
cerrarParentesis
puntoYcoma

P (2)

B (25)

S (19)

 alert
 abrirParentesis

E (31)

 R (33)

 U (36)

 V (37)

 V (39)

 W (40)

 identificador

 opAritmetico1

 W (40)

 identificador

 cerrarParentesis

 puntoYcoma

P (2)

B (25)

S (17)

 identificador

 asignacion

E (31)

 R (33)

 U (35)

 U (36)

 V (39)

 W (40)

 identificador

 opRelacional2

 V (39)

 W (40)

 identificador

 puntoYcoma

P (2)

B (23)

 if

 abrirParentesis

E (31)

 R (33)

 U (36)

 V (39)

 W (40)

 identificador

 cerrarParentesis

 S (17)

 identificador

asignacion
E (31)
R (33)
U (36)
V (38)
V (38)
V (37)
V (37)
V (39)
W (40)
identificador
opAritmetico1
W (43)
cteEntera
opAritmetico1
W (40)
identificador
opAritmetico2
W (43)
cteEntera
opAritmetico2
W (41)
abrirParentesis
E (31)
R (33)
U (36)
V (38)
V (38)
V (39)
W (43)
cteEntera
opAritmetico2
W (40)
identificador
opAritmetico2
W (43)
cteEntera
cerrarParentesis
puntoYcoma
P (4)
lambda

Prueba 4

```
let numero x;
```

Errores

Error Sintactico en la linea:1

 @Usuario: Se esperaba: number,boolean,string

 @Internal: Error en el token: <identificador, 0>

Prueba 5

```
alert (msg)
```

Errores

Error Sintactico en la linea:1

 @Usuario: Se esperaba: ;

 @Internal: Error en el token: <EOF, >

Prueba 6

```
do{
```

```
    alert(mensaje);
```

```
}while(valor = 1);
```

Errores

Error Sintactico en la linea:3

 @Usuario: Se esperaba: (,||,&&,==,!!=,+,- o nada

 @Internal: Error en el token: <asignacion, >