

ejercicios semana 1 y 2

convertir en lexemas y tokens

```
/*uso del while con ejemplo*/
int x; int y;
x = read();
y = read();
while(x!=y){
    if(y<=x){
        y=y-x;
    } else {
        x = x-y;
    }
}
/*salida de resultado*/
write("resultado: ");
write(x);
```

lexema	token
/*uso del while con ejemplo*/	comentarios
int	palabra_reservada
x	identificador
;	punto_coma
int	palabra_reservada
y	identificador
;	punto_coma
x	identificador
=	Comparación
read()	llamada_a_funcion
;	punto_coma

y	identificador
=	comparación
read()	llamada_a_funcion
;	punto_coma
while	palabra_reservada
(parentesis_izquierdo
x	identificador
!=	comparación
y	identificador
)	parentesis_derecho
{	llave_izquierda
if	condicional
(parentesis_izquierdo
y	Identificador
<=	comparación
x	identificador
)	parentesis_derecho
{	llave_izquierda
y	identificador
=	comparación
y	identificador
-	operador
x	identificador
;	punto_coma
}	llave_derecha

else	condicional
{	llave_izquierda
x	identificador
=	comparacion
x	identificador
-	operador
y	identificador
;	punto_y_coma
}	llave_derecha
}	llave_derecha
/*salida de resultado*/	comentario
write	llamada_funcion
(parentesis_izquierdo
"resultado: "	literal
)	parentesis_derecho
;	punto_coma
write	llamada_funcion
(parentesis_izquierdo
x	identificador
)	parentesis_derecho
;	punto_coma

```

float cuadradolimitado(float x)
{
    /*devuelve x al cuadrado, pero nunca más de 100*/
    float limiteinferior = -10.0;
    float limitesuperior = 10.0;
    return (x<=limiteinferior || x>= limitesuperior) ? 100: x*x;
}

```

lexema	token
float	palabra_reservada
cuadradolimitado	Identificador
(parentesis_izquierdo
float	palabra_reservada
x	identificador
)	parentesis_derecho
{	llave_izquierda
/*devuelve x al cuadrado, pero nunca más de 100*/	comentario
float	palabra_reservada
limiteinferior	identificador
=	comparacion
-10.00	numero
;	punto_coma
float	palabra_reservada
limitesuperior	identificador
=	comparacion
10.0	numero

;	punto_coma
return	palabra_reservada
(parentesis_izquierdo
x	identificador
<=	comparacion
limiteinferior	identificador
	operador
x	identificador
>=	comparacion
limitesuperior	identificador
)	parentesis_derecho
?	operador
100	numero
:	operador
x	id
*	operador
x	identificador
;	punto_coma
}	llave_derecha