**Luis Emilio Abreu ------------ 100015739**

**4. Consideraciones sobre la altura de los nodos.**

se necesitará tener acceso a la altura de cada nodo del árbol en tiempo constante. Dado que una función para hallar la altura de un nodo dado en un árbol tendrá un tiempo de ejecución de O(log(n)), por lo tanto, es necesario almacenar una variable altura en cada nodo e irla actualizando en las inserciones y eliminaciones que se efectúen sobre el árbol. Como nos muestra el siguiente código.

int altura(AVLTree \*t)

{

if(es\_vacio(t))

return -1;

else

return max(altura(izquierdo(t)), altura(derecho(t)));

}

Esto nos dice que cada vez que insertemos o eliminemos un nodo se debe almacenar su altura tanto a la izquierda como a la derecha.

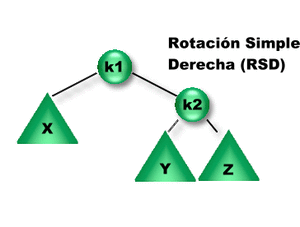
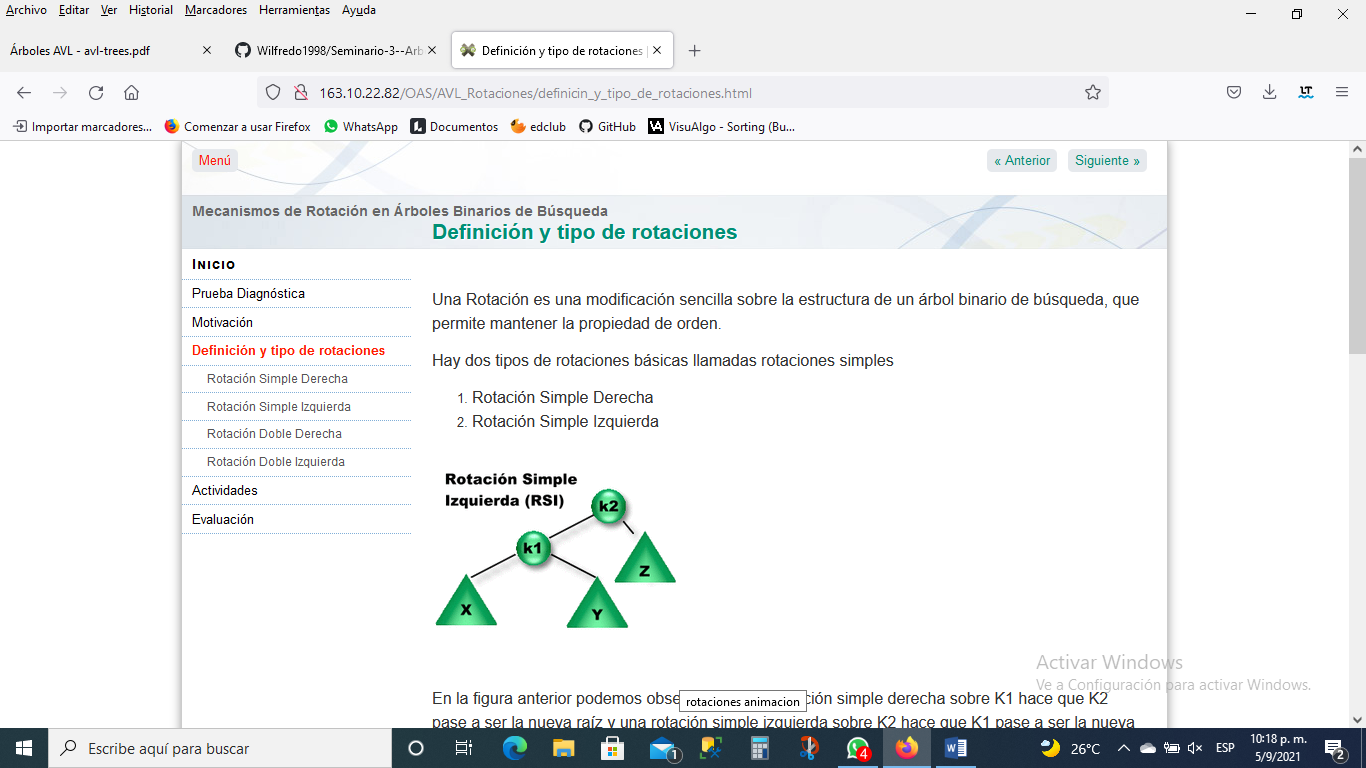
**5. Rotaciones simples.**

Una Rotación es una modificación sencilla sobre la estructura de un árbol binario de búsqueda, que permite mantener la propiedad de orden.

Hay dos tipos de rotaciones básicas llamadas rotaciones simples

* Rotación Simple Derecha
* Rotación Simple Izquierda

Como podemos ver en la siguiente imagen



Como podemos observar una rotación simple derecha sobre K1 hace que K2 pase a ser la nueva raíz y una rotación simple izquierda sobre K2 hace que K1 pase a ser la nueva raíz. En ambos casos los subárboles de K1 y K2 se reacomodan respetando la propiedad de orden.